Federation University

FedUni ResearchOnline

https://researchonline.federation.edu.au

Copyright Notice

This is the published version of:

Rashid, M. M., Kamruzzaman, J., Hassan, M. M., Shahriar Shafin, S., & Bhuiyan, M. Z. A. (2020). A Survey on Behavioral Pattern Mining From Sensor Data in Internet of Things. IEEE Access, 8, 33318–33341.

Available online at: https://doi.org/10.1109/ACCESS.2020.2974035

Copyright @ IEEE 2020. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY 4.0) (http://creativecommons.org/licenses/by/4.0/). The use, distribution or reproduction in other forums is permitted, provided the original author(s) or licensor are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 12, 2020, accepted February 4, 2020, date of publication February 14, 2020, date of current version February 26, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2974035

A Survey on Behavioral Pattern Mining From Sensor Data in Internet of Things

MD. MAMUNUR RASHID[1], (Member, IEEE), JOARDER KAMRUZZAMAN[2], (Senior Member, IEEE), MOHAMMAD MEHEDI HASSAN[3], (Senior Member, IEEE), SAKIB SHAHRIAR SHAFIN[4], AND MD. ZAKIRUL ALAM BHUIYAN[5], (Member, IEEE)

[1]School of Engineering and Technology, CQUniversity Melbourne, Melbourne, VIC 3000, Australia
[2]School of Science and Information Technology, Federation University Australia, Ballarat, VIC 3350, Australia
[3]College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia
[4]Department of Electrical and Electronic Engineering, Islamic University of Technology, Gazipur 1704, Bangladesh
[5]Department of Computer and Information Science, Fordham University, New York City, NY 10458, USA

Corresponding author: Mohammad Mehedi Hassan (mmhassan@ksu.edu.sa)

This work was supported by the King Saud University, Riyadh, Saudi Arabia, through Researchers Supporting under Project RSP-2019/18.

ABSTRACT The deployment of large-scale wireless sensor networks (WSNs) for the Internet of Things (IoT) applications is increasing day-by-day, especially with the emergence of smart city services. The sensor data streams generated from these applications are largely dynamic, heterogeneous, and often geographically distributed over large areas. For high-value use in business, industry and services, these data streams must be mined to extract insightful knowledge, such as about monitoring (e.g., discovering certain behaviors over a deployed area) or network diagnostics (e.g., predicting faulty sensor nodes). However, due to the inherent constraints of sensor networks and application requirements, traditional data mining techniques cannot be directly used to mine IoT data streams efficiently and accurately in real-time. In the last decade, a number of works have been reported in the literature proposing behavioral pattern mining algorithms for sensor networks. This paper presents the technical challenges that need to be considered for mining sensor data. It then provides a thorough review of the mining techniques proposed in the recent literature to mine behavioral patterns from sensor data in IoT, and their characteristics and differences are highlighted and compared. We also propose a behavioral pattern mining framework for IoT and discuss possible future research directions in this area.

INDEX TERMS Association rules, behavioral patterns, data mining, frequent pattern, Internet of Things, knowledge discovery, wireless sensor networks.

I. INTRODUCTION

In recent years wireless sensor networks have demonstrated promising applications in many diverse areas including precision agriculture, environment monitoring, industrial automation, asset management, remote health monitoring, and military applications [1]–[3]. The push towards building smarter and smaller sensor devices, and the low-cost deployment of sensors have given rise to large scale and dense WSNs to create diverse smart city services ranging from traffic management, emergency incident management to public safety [4]–[7]. The increasing adaptability and simplicity of deployment of sensor networks will continue to widen their applications in many other diverse areas. The Internet of Things (IoT) is a push towards the integration of data providers with end-users of the Internet and various communication networks [8]. The vision of IoT will lead to an information-rich connected world [9] and WSNs in an interconnected way form its basic building blocks. The IoT permits the interconnection of different types of everyday objects that have identities and physical and virtual attributes, and can be seamlessly integrated with the Internet, enabling direct user involvement in operations of the integrated equipment [10]. The National Intelligence Council (NIC) [11] states that "*By 2025 Internet nodes may reside in everyday things like food packages, furniture, paper documents, and more*".

The associate editor coordinating the review of this manuscript and approving it for publication was Md. Fazlul Kader.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see http://creativecommons.org/licenses/by/4.0/

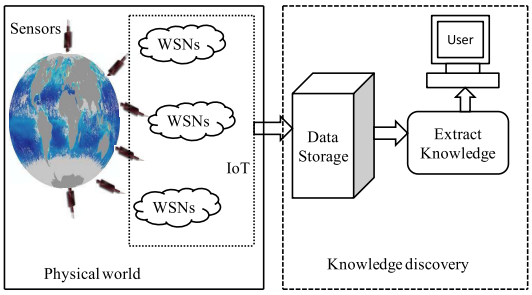


FIGURE 1. Knowledge extraction from the physical world.

These everyday things along with numerous distributed large scale WSNs will generate huge amounts of data, especially in the envisaged IoT scenario, where terabytes of data are expected from billions of sensors [12]. These voluminous data from IoT worth little in practical terms unless useful knowledge can be mined from the data stream. However, this presents significant and new challenges to the knowledge discovery process. A schematic illustration of knowledge extraction from WSNs and the IoT is shown in Figure 1.

In traditional databases, knowledge is extracted from the massive amount of collected data through the discovery of useful patterns that exhibit some important information about the system, which is often vital in making critical business and management decisions. Knowledge discovery in databases (KDD) is used in many fields such as banking, retail market, manufacturing, machine condition monitoring, health care system, marketing and science data acquisition. Data mining, the core of KDD is an iterative and interactive process of finding novel, substantial and valuable patterns and models in large datasets. The models are utilized for the comprehension of phenomena from the available information and make predictions for the future.

In IoT applications comprising large scale WSNs, data mining techniques are used to extract behavioral patterns from a continuous and rapid flow of data stream. However, here the problem is to store the whole data and process them immediately. To handle such high speed data, data mining models need to be fast. Existing traditional data mining techniques [13], [14] are not able to process the huge amount of sensor data in an acceptable time because of its high dimensionality and distributed nature. Moreover, traditional data mining models are centralized which suffers from high computational cost due to data accumulated at a central site. However, sensor data which flow continuously in the systems at varying rates and in massive quantity in the IoT environment incur high storage cost. Thus it is impossible to store the entire dataset or to scan it multiple times for mining purposes as some of the traditional mining algorithms require to scan through the whole dataset multiple times for them to work. Therefore, to process such stream data, it is essential to develop data mining techniques that can handle sensor data in a single pass, multidimensional, and real time manner.

It should be noted that in both WSN and IoT, sensors are the individual data sources. In WSN, sensors send their sensed data to the cluster heads (several sensors form a cluster) which then sends the data to the sink/gateway which is connected to an application through the business's own network or the Internet. In the IoT, sensors may be connected directly to the Internet and WSNs connecting to the Internet. Therefore, once data are received from the source sensor, they are the same for WSNs and IoT. However, depending on applications, IoT data are most likely to be larger in volume and geographically distributed over a larger area than WSN, likely to have real-time significance, and exhibit more variation and experience a greater change in nature over time. WSNs have naturally progressed towards the IoT and they are the main building blocks of the IoT. Therefore, data mining techniques first proposed for WSN gradually progressed toward IoT. Discussion on data mining techniques for IoT can not be done without discussing the techniques for WSN. Therefore, for the systematic presentation of various approaches proposed in the literature, our discussion on data mining techniques for WSN and IoT are presented together in the subsequent sections.

The knowledge discovery process in IoT (KDIoT) using data mining techniques suitable to handle large scale and stream sensor data can serve as an effective tool to enhance the quality of service (QoS) of many applications and the performance of networks [15]. The following types of knowledge can be mined using KDIoT:

1) Patterns extracted from the sensor network data for environment monitoring [16]–[18];
2) Behavioral patterns discovering sensor behavior from the meta-data describing them.

The knowledge discovery process, in general, requires a progression of steps, including domain understanding, knowledge definition, data preparation and data mining [19], [20]. The 'evolution' from KDD to KDIoT necessitates enhancing most of the methods in KDD. In addition, new techniques specifically for KDIoT need to be developed [21]. The extraction of behavioral patterns from sensor data is a complex process. It requires extensive efforts to extract the patterns that describe sensor activities in a WSN. Regarding this matter, several issues must be addressed: i) various aspects of behavioral patterns, ii) the impacts of these patterns on WSNs operation, and iii) the challenges that the knowledge discovery process face when generating these patterns.

Most of the existing survey papers [22]–[24] on sensor data mining mainly focused on outlier detection from the WSNs. Clustering based survey papers [25], [26] present architecture and management of the sensor network instead of information discovery. In [27], a classification based survey paper is presented where the conventional classification techniques are evaluated over the data stream. Another survey paper on data mining techniques [28] provides an overview of how traditional data mining techniques are revised and improved to enhance the performance of WSNs. Some other works [29], [30] highlighted the data mining techniques on the IoT environment. To the best of our knowledge, there does not exist any comprehensive survey on behavioral pattern

mining techniques applied to sensor data in IoT applications. This paper presents a survey of recent research works on behavioral patterns in sensor data to help researchers easily locate some seminal works in this area. The main purpose of this survey is to introduce readers with a brief overview of the existing methods, their key characteristics, merits and limitations along with the future research trends and challenges on this topic. Here, we have highlighted the challenges of mining behavioral patterns in such scenarios, critically analyzed the existing techniques and proposed a knowledge-based framework to mine them efficiently in an IoT environment.

The contributions of this paper are summarized as follows:

- Presenting a brief overview of past survey papers on behavioral pattern mining on WSN/IoT data, and identifying the gap in the survey literature and underpinning the need for the current survey.
- Identifying the challenges of extracting behavioral patterns from WSN and IoT data.
- Providing detail discussion and critical analyses of the existing behavioral pattern mining techniques developed for WSN and IoT.
- Proposed a knowledge-based framework to overcome the limitations of existing techniques and mine patterns offline as well as online.
- Identifying future research challenges on this topic and outlining directions on how those challenges can be addressed.

The paper is organized in the following sections. Section 2 discusses the related survey papers on data mining techniques in WSN/IoT. Section 3 discusses the applications of behavioral patterns in IoT. Section 4 provides the fundamentals of WSNs and the main challenges of behavioral patterns mining from IoT data. Section 5 presents a technique-based taxonomy to categorize the existing behavioral patterns mining techniques developed for IoT as their key features. Sections 6 presents a description of the current behavioral pattern mining techniques proposed to mine from IoT data, analysing their strengths and limitations. Section 7 discusses some open research issues in this regard and finally, Section 8 presents concluding remarks.

II. RELATED SURVEY WORKS

Only a limited number of survey papers have been published in the literature that considered data mining on WSN and/or IoT data. In Table 1, we list the notable survey papers on data mining in WSN/IoT and summarize the scope of their works in brief. These papers can be categorised into two groups.

A. SURVEY PAPERS THAT DID NOT CONSIDER BEHAVIORAL PATTERNS

In [31], Chen et al. provided an overview of traditional data mining functionality such as classification, clustering, association, anomaly detection and time series analysis, and describe different applications where these functionalities are used. They also proposed a big data mining system for IoT. In [30], Ahsan and Bari reviewed the impact of big data in IoT by using its protocols and architecture. Different techniques for verifying these protocols and their security factors are also discussed. Marjani *et al.* [32] explored the existing notable works on big IoT data analytics and discussed the association of big data and IoT data analytics, following which they proposed a framework for big IoT data analytics. In [29], a brief review of data mining techniques for IoT has been presented and a data analytics reference model to discover meaningful information from the IoT environment has been explored. In [33], Shadroo and Rahmani reviewed 44 research articles to explore the data mining techniques on big IoT data which they classified into three groups, namely, architecture & platform, framework and applications. The above survey papers concentrated on how data mining techniques are used to extract the hidden knowledge from IoT [31], [33], how big data analysis can be performed in the IoT environment [30] and discussed a reference model for data analysis [29], [32]. However, these works do not consider behavioral patterns.

B. SURVEY PAPERS THAT CONSIDERED BEHAVIORAL PATTERNS

In [28], Mahmood et al. provided an overview of how traditional data mining techniques (frequent mining, sequence mining, clustering and classification) were modified and improved to enhance the performance of WSNs. However, this survey only considered frequent patterns-based behavioral patterns and did not explore interestingness-based patterns. In [34], Tsai et al. reviewed the existing data mining techniques for IoT environments where they explored these techniques for the infrastructure as well as services of IoT and demonstrated pattern discovery from smart home applications. However, they did not provide details on how these patterns can be used to recognize human activities. In [35], data mining techniques used in the industrial IoT (IIoT) were briefly reviewed. The present and future trends of IoT on the aspects of data analytics were also discussed. The temporal management of large-scale RFID applications (TMS-RFID) and intelligent RFID examples was investigated where frequent pattern-based data mining techniques have been used. In [36], Braun et al. reviewed the data mining methods to discover patterns from big IoT data with fog computing. To mine patterns, they proposed two methods: firstly, pattern mining through local networking services and secondly, patterns mining on local IoT devices. They also presented a case study of real-life applications for urban analytics based on frequent patterns using the second method. In these methods, the computations are performed near to the end-users which can reduce the latency and bandwidth of the network and enhance the network security and reliability. However, they did not investigate the data collection mechanism and interestingness based behavioral pattern mining.

C. NEED FOR A NEW SURVEY PAPER

Most of the existing survey papers (e.g., [29], [30]) mainly focused on general data mining techniques in sensor networks or IoT, and only a few of them covered some aspects of

TABLE 1. Summary of survey works on data mining in WSN/IoT.

Survey articles	Year	Scope	Covered behavioral patterns mining
Data mining techniques in WSN [28]	2013	Fundamental of data mining in WSNs, data mining techniques such as classification, clusterting, sequential pattern and frequent patterns mining in WSN	Yes, but only considered frequent pattern-based behavioral patterns
Data mining for IoT [34]	2014	Mining algorithm from IoT	Yes, but only considered frequent pattern-based behavioral patterns
Data mining for IoT [31]	2015	Data mining functionality, open research issues on IoT data mining	No
Data analysis issue in IoT [35]	2015	The impact of cognitive capabilities and IoT data analytics	Yes, but only considered frequent pattern-based behavioral patterns
Big data analysis and IoT [30]	2016	The role of big data in IoT	No
Big IoT data analytics [32]	2017	Overview of Big data and IoT, relationship between big data and IoT	No
Data analytics in IoT [29]	2018	Data analytic architecture for IoT	No
Big data and data mining in IoT [33]	2018	IoT big data and IoT data mining	No
Pattern mining over Big [36]	2019	Pattern mining from IoT using Fog Computing	Yes, but did not consider the issues below : - real IoT environment - details of data collection mechanism - interestingness based behavioral patterns

frequent pattern-based behavioral patterns (e.g., [28] (2013), [35] (2015)) but did not explore in details the evolving IoT scenario and its applications. Moreover, in the last decade, many works have been published focusing on the interestingness based behavioral patterns, but there exists no survey work in literature solely on this topic. The impact of behavioral patterns in real-world scenarios such as IoT-based smart city, industry and other areas is very significant (please see the applications in Section 3 for more details). There are many smart city services being launched around the world and such services have increasing value to the communities, businesses, scientific bodies and governments around the world. Behavioral pattern mining form these applications will improve the quality of smart cities and other IoT based services and lead to better real-time insights and identification of correlated events for economically efficient resource management by the local/state governments and businesses. These initiatives and extensive research for efficient and robust sensor data mining techniques in real time have created the need for an extensive survey paper on this topic focusing on behavioral patterns.

III. APPLICATIONS OF BEHAVIORAL PATTERN MINING IN IOT

Mining behavioral patterns from IoT could be highly useful in diverse areas such as industrial IoT, smart city services and other applications that need real-time monitoring of the physical environment such as smart building, remote patients and infrastructures, and analysis of collected data for useful knowledge. Below some application areas of behavioral patterns mining are discussed.

A. INDUSTRIAL IOT

In industrial IoT (IIoT), behavioral patterns can be used to predict the source of a future event which, in turn, can identify the faulty nodes of the network [37]. For example, if we are expecting an event from a node but not receiving any such report from that node, it suggests the node may be malfunctioning. Moreover, behavioral patterns also can be used to predict the next event. The reason is that behavioral patterns are able to reveal a chain of related events, which is very important in the industrial perspective. For example, in the industry, an error in a particular process may lead to other errors/faults [38].

B. HEALTH CARE

IoT has brought up new opportunities in health care, especially in eHealth. A piece of medical equipment when connected to the Internet can easily send important data from various patients to a central health care database. Mining behavioral patterns from these data can discover vital knowledge such as patients' symptoms and trends and can make remote care more effective using the extracted knowledge [39]. In this way, it helps patients' control over their diseases and treatment plans. Behavioral patterns are also useful to the health service providers in many ways for better planning of patient caregiving such as better and targeted monitoring of patient's physical condition and analysis of medical billings [40].

C. SMART CITY SERVICES

Recent research on light-weight service mashup middleware for IoT applications that allow the physical things seamless integration into the Web. This help the easy development and deployment of IoT-based smart city applications [41]. In these applications, discovering behavioral patterns can significantly improve services such as transportation, energy consumption and security. By using the knowledge gained through such patterns, it is possible to predict which citizens are going to leave the city based on which factors of city

services, and then the city authority can work to improve on these factors in the future [31]. Another highly important application is to identify the crime hot spots based on crime data which can be collected from IoT visual sensor data [31], [42].

D. IOT RESOURCE MANAGEMENT

Behavioral patterns can identify a set of temporally correlated sensors. This information is useful to resolve the unwanted impacts such as missed readings due to the sometimes unreliable wireless communications. By knowing the correlated set, it is possible to identify which sensors can be switched to the sleep mode to conserve energy without compromising the network coverage area. In these ways, behavioral patterns can be utilised for better resource management and safety assurance in IoT [43], [44].

IV. MAJOR CHALLENGES IN RELATION TO KNOWLEDGE DISCOVERY FROM IOT

Sensing environment, monitoring activities or detecting an event through identifying a change in the state within the region of interest remains the fundamental tasks of WSNs, the primary building blocks of the IoT [45]–[47]. Detecting these events or monitoring activities is feasible by processing and analysing sensor data obtained from sensor nodes [48]. According to Watanabe [49] a pattern is ''opposite of chaos''. Catania *et al.* [50] defines a pattern as ''a compact and rich in semantics representation of raw data''. In this paper, we define a pattern as the set of sensor nodes which is extracted from the meta-data describing sensor behaviors.

Though the traditional data mining techniques have matured over the years, they cannot be directly used to handle sensor data from IoT for the following challenges [28]:

- **Resource constraints:** Due to their construct, most sensor nodes have limited computational power, memory, energy, and bandwidth. These constraints impose new challenges to devise efficient and accurate data mining techniques for WSNs while utilizing minimum resources.
- **Faster stream of large data:** Sensor data in many IoT applications may arrive faster than the speed at which those data can be mined properly. Here, the challenge is how data mining techniques can manage the huge volume of fast, continuous and changing data streams while supporting user interaction at the same time.
- **Real-time mining:** The occurrence of new events in a geographically distributed WSN or IoT will generate a stream of data, which may exhibit different characteristics depending on the nature of the events. Consequently, the mining results based on the new data are likely to be significantly different from the old data, and in critical applications, the new data need to be mined in real time to discover new patterns. This poses a new challenge to mine distributed data streams in the IoT in real time.
- **Capturing changes over time:** The IoT data monitoring environmental phenomenon will experience change over time. Therefore, only mining results from the data are not enough, rather it is highly important to capture any significant change in the mining results over time. Here, the research issue is to model this change in the mining technique with relevance to a particular application.
- **Data transmission:** Since WSNs suffer from resource limitation in terms of bandwidth, transmission of generated data directly is ofetn infeasible and will cost unnecessary energy, even with the development of low power wide area network technologies. After extracting patterns locally from a WSN, the mining results are transmitted instead. Research needs to be done to efficiently represent data and discovered patterns so that they can be transmitted using low bandwidth.
- **Complexity of deployment:** In many cases, heterogeneous sensors are needed to be deployed in harsh, inaccessible and dynamic environments. Mobile sensors are also deployed to cover any void area or to provide k-fold coverage for better event detection accuracy [51]. The complexity of data mining techniques is increased by such dynamic environments.

To overcome the above issues, research works have focused on modifying the traditional data mining approaches as well as devising new approaches suitable for IoT. Some works have focused on discovering patterns from the sensed data stored in a central database [52]–[55] while others have focused on extracting patterns from sensor nodes through discovering association rules (SARs) among them [56]–[58]. A sensor association rule can be expressed as $(s_1, s_2 \rightarrow s_3, 75\%, \lambda)$. This is interpreted as, if sensors s_1 and s_2 have reported events, then there exits 75% chance that sensor s_3 will detect an event within λ units of time.

Several challenges in discovering the behavioral patterns and their aspects from WSN/IoT sensor data and networks require major modification to the traditional knowledge discovery process. These challenges include:

1) The need for an appropriate formulation to discover a behavioral pattern that identifies the knowledge in terms of sensor terminologies and maintains the downward closure property (i.e., if a pattern is infrequent, then all of its super patterns are also infrequent).
2) In most applications, sensor networks produce huge volumes of data within a short duration of time. Therefore, compact data structures are required to store the meta-data.
3) To generate behavioral patterns, it is necessary to consider every conceivable relation that can be characterized by sensor nodes. These relations must be checked against the meta-data to recognize the relations of interest to respective applications. Efficient algorithms are required to mine these behavioral patterns in the shortest amount of time and with the smallest memory cost.
4) Most of the existing sensor data mining techniques have been proposed for single processor machines. To process huge sensor data with limited resources is

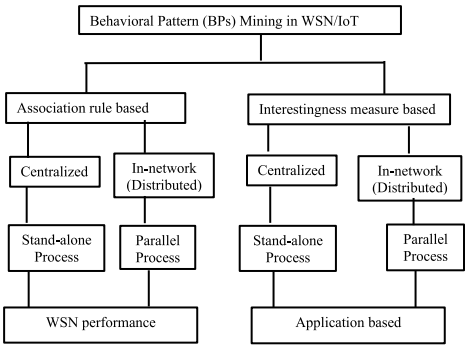


FIGURE 2. Taxonomy framework for behavioral patterns mining techniques designed for WSNs, leading to IoT.

a challenging task. Therefore, the development of a suitable parallel and distributed model [59], [60] for large scale sensor data mining is a major challenge.

V. TAXONOMY FRAMEWORK FOR BEHAVIORAL PATTERNS MINING TECHNIQUES FOR IOT

Data mining techniques are mainly divided into four categories: (i) clustering (ii) classification (iii) sequential pattern mining and (iv) association rules. Cluster-based methods are K-mean, hierarchical and data correlation-based techniques based on the distance among the data points. Classification-based methods are the decision tree, rule-based, nearest neighbor and support vector machine. Most of the existing sequential pattern mining and association rules mining techniques in WSN, and hence in IoT, are adapted from traditional Apriori [13] and frequent pattern (FP) growth-based techniques [14]. Figure 2 shows the taxonomy of the mining framework in WSNs, leading to IoT. In this survey paper, our main focus is on mining behavioral patterns (BPs) using association rules mining techniques. Association rules can be further classified based on the following items:

- **Data processing:** Association rules mining techniques for WSNs can be classified based on data processing location: Centralized and Distributed. In a centralized method, data from the entire network is stored in a central site for further analysis. In this case, the initial data reduction is performed in the central site [56], [58]. On the other hand, the in-network method considers the limited resource of sensor nodes and performs some extra computation in the nodes to limit the message and communication energy during transferring the data to the central site. In a nutshell, in the centralized method, the data are transmitted to the central site without any optimization from the sensors, while in the in-network method [52], [53] nodes optimize the messages sent to the central cite taking account of resource limitations.
- **Pattern nature:** Mining the data from the central or decentralized method based on pattern nature can be classified as: Frequent patterns and Interestingness measure based patterns. Support metric-based sensor frequent patterns use the occurrence frequency of patterns as a criterion. However, support metric based behavioral patterns, frequent pattern mining from real sensor data is not simple. The rule depends on a constraint known as *minimum support threshold* (*min_sup*). The threshold specifies the minimum lower bound for the support of the resulting association rules. With a high *min_sup* value, only high-value knowledge in few rules is generated. If the value of *min_sup* is set low, a large number of rules are generated, but only a few of them are informative. Since large scale WSNs generate huge amounts of data, it is essential to use the appropriate 'interestingness' measure to discover sensor behavioral patterns that have strong correlations among the data [43]. On the other hand, the support measure value used in the traditional mining algorithm cannot resolve practical issues. For instance, in a specific time slot, a sensor may trigger multiple times. By analyzing these non-binary trigger values one can extract more important knowledge from the sensor data [61]. Another criterion to identify the 'interestingness' of frequent patterns is the shape of occurrence, i.e., whether their occurrence is regular, irregular, or mostly in specific time intervals in the sensor database [62].
- **Computation:** How the computation is performed to mine frequent patterns and interestingness patterns can be classified as: stand-alone process and parallel process. Stand-alone process only considers single processor and main memory-based machine for frequent pattern [56] and 'interestingness' patterns mining [63]. As discussed earlier, resource constraints in sensor nodes and networks present big computational challenges for real-time mining of large sensor data. Therefore, to mine such kind of large data, more efficient approaches such as parallel and distributed techniques (besides serial approach) are needed [61], [64].
- **Specific problem solving:** The behavioral patterns can be classified based on solving a specific problem such as WSNs performance related and IoT applications related. Since, sensor nodes have resource constrained, resource aware techniques are essentials to maximize the performance of WSNs. On the other hand, IoT applications need to be fault tolerant, scalable, robust and accurate, and often require abundant use of energy, communication, and redundancies.

VI. BEHAVOIRAL PATTERN MINING FROM IOT

This section represents a formal definition of the fundamental concepts necessary to deal with behavioral sensor patterns in the IoT.

A. PRELIMINARY

Let $S = \{s_1, s_2, \ldots, s_p\}$ be the set of sensors deployed in a WSN and the time be divided into equal-sized slots $t = \{t_1, t_2, \ldots, t_q\}$ such that $t_{j+1} - t_j = \lambda, j \in [1, q-1]$ where λ is the slot size. $T_{his} = t_q - t_1$ is the historical period

of the data defined during the data extraction process. A set $P = \{s_1, s_2, \ldots, s_n\} \subseteq S$ is called a pattern of sensors.

An epoch is defined as a tuple $e(e_{ts}, Y)$ where Y is a pattern of the event-detecting sensors that report events within the same time slot and e_{ts} is the epoch's time slot. A sensor database SD is a set of epochs $E = \{e_1, e_2, \ldots, e_m\}$ with $m = |SD|$, i.e., total number of epochs in SD. If $X \subseteq Y$, it is said that X occurs in e and is denoted as $e_j^X, j \in [1, m]$. Let $E^X = \left\{e_j^X, \ldots, e_k^X\right\}$, where $j \leq k$ and $j, k \in [1, m]$ be the ordered set of epochs in which pattern X has occurred in SD. Let e_s^X and e_t^X, where $j \leq s < t \leq k$ be the two consecutive epochs in E^X. The number of epochs or time difference between e_t^X and e_s^X, can be defined as a period of X, say p^X. Then a period of X, $p^X = \left\{e_t^X - e_s^X\right\}$. Let $P^X = \left\{p_1^X, p_2^X, \ldots, p_s^X\right\}$ be the set of periods for pattern X. For simplicity in period computation, we assume the first and last epochs (say, e_f and e_l) in SD are identified as *null* with $e_f = 0$ and e_m with $(e_l = e_m)$, respectively. An example of a sensor database, *SD* is illustrated in Table 2.

TABLE 2. An example of a sensor database (*SD*).

TS	Epoch
1	$s_1\ s_5\ s_6$
2	$s_1\ s_2\ s_3\ s_4\ s_7$
3	$s_1\ s_2\ s_4\ s_7$
4	$s_2\ s_5\ s_6\ s_7$
5	$s_1\ s_2\ s_3\ s_4\ s_7$
6	$s_1\ s_2\ s_4\ s_5$

Definition 1 (support of pattern X in SD): The support, i.e., occurrence frequency of the pattern X in *SD* is defined to be the number of epochs in *SD* that support it, i.e., $sup(X) = |E(E_{ts}, Y)|X \subseteq Y|$. The maximum sensor support of the pattern X can be defined as, $Max\_sensor\_Sup(X) = Max(Sup(s_j)|\forall s_j \in X)$. The interestingness measure *all-confidence* denoted by α of a pattern X is defined as follows:

$$\alpha = \frac{Sup(X)}{Max\_sensor\_Sup(X)} \tag{1}$$

Definition 2 (Associated Pattern): A pattern is called an associated pattern, if its *all-confidence* is greater than or equal to the given minimum *all-confidence* threshold, denoted as *min_all_conf*.

Definition 3 (Regularity of pattern X): Let for a E^X, P^X be the set of all periods of X i.e., $P^X = \left\{p_1^X, p_2^X, \ldots, p_N^X\right\}$, where N is the total number of periods in P^X. Then the average period value of pattern X is represented as, $\bar{X} = \sum_{k=1}^{N} \frac{P_k^X}{N}$ and the variance of periods is represented as $\sigma^X = \sum_{k=1}^{N} \frac{(P_k^X - \bar{X})^2}{N}$.

Definition 4 (Regularly frequent sensor pattern): A pattern is called a regularly frequent pattern if it satisfies both of the following two conditions: (i) its support value is no less than a user-given minimum support threshold, say, *min_sup* and (ii) its regularity is no greater than a user-given maximum regularity threshold, say, *max_var*.

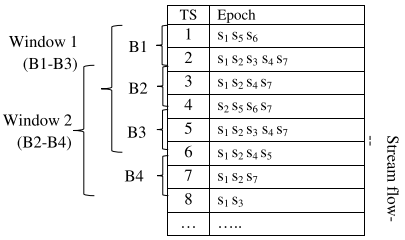


FIGURE 3. Sensor data stream (SDS).

On the other hand, sensor data stream (SDS) is a continuous, unbounded and ordered sequence of data. Therefore, it is impossible to maintain all the elements of a sensor data stream in a tree over a long period of time. Moreover, old information may become obsolete and recent information may become important from a knowledge discovery point of view. To handle such a scenario, the sliding window model is used to keep track of recent epochs where epochs are maintained in a batch-by-batch manner. Fig. 3, depicts a *SDS* where a window of six epochs consists of three batches. However, to facilitate such a model, appropriate data observation window size needs to be determined.

A sensor data stream *SDS* can formally be defined as an infinite sequence of epochs (e), i.e., $SDS = [e_1, e_2, \ldots, e_n)$, where $e_r, r \in [1, n]$ is the r-th arrived epoch. Each epoch is a tuple $e(e_{ts}, Y)$, where e, e_{ts} and Y are defined earlier. A window W can be referred to as a set of all epochs between the r-th and s-th $(s > r)$ epochs and the size of W is $|W| = s - r$. If there are M epochs and N batches in a W, then each batch consists M/N epochs; hence, the size of each batch is $|M/N|$. Here, the window slides batch-by-batch, i.e., sliding adds newer batch and removes older batch from the current window.

Definition 3 (support of pattern X in W): The support of a pattern X in a W, denoted as $Sup_w(X)$ is the number of epochs in W that contain X. Therefore, a pattern is called frequent in W, if its support is no less than *min_sup*, such that $0 \leq min\_sup \leq |W|$.

Definition 5 (all-confidence of pattern X in W): The *all-confidence* of a pattern X in a W, denoted by $\alpha_w(X)$ is defined as follows:

$$\alpha_w = \frac{Sup_w(X)}{Max\_sensor\_Sup_w(X)} \tag{2}$$

Definition 6 (associated sensor pattern X in W): A pattern X is called an associated sensor pattern in W, if its *all-confidence*, $\alpha_w(X)$ is greater than or equal to the given minimum *all-confidence* threshold in W.

Definition 7 (Regularity of pattern X in W): Let for a ${E_W}^X$, ${P_W}^X$ be the set of all periods of X in W. Then the average period value of pattern X is represented as, $\bar{X_W} = \sum_{k=1}^{N_W} \frac{P_k^X}{N_W}$ and the variance of periods is represented as $\sigma^{X_W} = \sum_{k=1}^{N_W} \frac{(P_k^X - \bar{X_W})^2}{N_W}$.

Definition 8 (Regularly frequent sensor patternX in W): A pattern X is called a regularly frequent pattern in W,

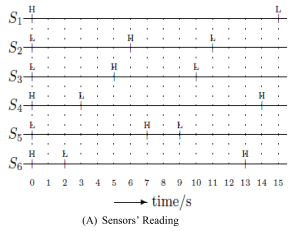
if its support value is no less than a user-given minimum support threshold, say, min_sup_w and its regularity is no greater than a user-given maximum regularity threshold, say, max_var_w in W.

B. ASSOCIATION RULE BASED BPS MINING TECHNIQUES

In this section, we discuss association rule based mining techniques that employ distributed as well as centralized processing scheme. In a distributed processing scheme, major processing tasks are shifted towards individual sensors to build a local model.

Loo *et al.* [52] considers the tasks of mining associations among sensor data values that coexist temporally in a large-scale WSN. Their technique relies on a data model where continuous sensor readings are quantized to assume values from a finite set of discrete values. The data model stores readings reported by sensors and presents them in a way so that Loosy counting algorithm [65] can be applied for one-pass analysis of the data. Sensor readings are reported at regular intervals and snapshots from the sensors' readings are recorded only when any change in the readings is detected. A database then stores these snapshots as contexts. Taking snapshots at state changes reduces redundancy in the database. To overcome the problem of random state changes in sensor nodes, each context is associated with a weight value indicating the number of intervals for which this context is valid. To illustrate this model, we use the same example provided in [52]. Figure 4(A) (Figures A and B are redrawn from [52]) depicts the states of six sensor nodes during a period of 15 seconds. Each sensor takes a value from two possible states (High, H; Low, L). For instance, sensor s_2 exhibits state L, at time 0; and state H, at time 6. The first context is $\{(s_1, \text{H}), (s_2, \text{L}), (s_3, \text{L}), (s_4, \text{H}), (s_5, \text{L}), (s_6, \text{H})\}$, which is valid for two seconds before a state change occurs. Figure 4(B) shows the extracted database.

Their proposed data model represents the problem of mining associations among sensors' values where each possible sensor state is considered as an object; and a pattern is a set of sensors' states. For example (s_1 = L, s_4 = H) is one of the possible patterns. The support of the pattern is formulated by the total length of non-overlapping intervals in which the pattern is valid. To facilitate support counting, sensor data are represented by interval lists, where the interval list is a list of pairs containing the start time and end time for which the patterns are valid. For example, the interval list of the pattern s_5 = L is (i.e., IL(s_5 = L)) is [(0-7)]. A Loosy counting algorithm [65] is then used to generate frequent patterns. The advantage of using this algorithm is that it can make an online analysis with one pass of the database, which does not generate the exact frequent patterns (i.e., value sets). However, it gives a solution with a bounded error [65].



(A) Sensors' Reading

Transaction ID	S_1	S_2	S_3	S_4	S_5	S_6	weight
1	H	L	L	H	L	H	2
2	H	L	L	H	L	L	1
3	H	L	L	L	L	L	2
4	H	L	H	L	L	L	1

(B) The Database

FIGURE 4. Example of inter-stream mining.

Romer [53] proposed a method to mine spatial-temporal event patterns from the WSN data. Considering the distributed nature of WSNs, the authors devised an in-network data mining technique to mine frequent event patterns and their spatiotemporal relationships within the same network. In this way, only compact patterns need to be transmitted from the nodes to the sink, instead of row streams of data. This reduces the communication overhead. In this case, a sensor records the events detected within its certain distance. The distance may be expressed in terms of the number of hops or Euclidean distance. On the collection of the events, a sensor uses a mining technique to discover the patterns that satisfy the given parameters. The mining parameters for this approach include *min_sup*, *min_conf* , maximum scope, and maximum history. Each node collects the events from the neighbors within the maximum scope and keeps a history of their events for the duration of the maximum history. Every node then uses a mining algorithm to discover the frequent patterns of the form:

$$a_1\hat{a}_2\hat{} \ldots a_m \Rightarrow e[\text{min\_sup}, \quad \text{min\_conf}].$$

The above means that if all the predicates in the rule antecedent become true, then event E may occur at the node with support (S) and confidence (C). Each predicate in the rule antecedent is in the form $a_i = (e_i, d_i, t_i, n_i)$. a_i is true if and only if event e_i occurred n_i times at a distance d_i from the node and t_i time units before the occurrence of event e [53].

To adapt Romer's framework for the association mining problem, a quantization technique is utilized to quantize the continuous parameters like distance, time offset, and the number of occurrences of the events. For example, the distance parameter can be divided into two variables, near (between 0 and 5 meters), and far (between 5 and maximum scope). Nodes start collecting events from their neighbors at regular intervals. Each interval is called an epoch. Each node maintains a table of the number of events times the number of distance's partition columns. A cell corresponding to the column (e_i, d_i) is incremented once an event e, is received from a node within the distance d_i. The table contains one row for each possible epoch in the given maximum history and at the end of the historical period, the table is grouped and summed based on the time partitions. A context is then created for each epoch. Moreover, each possible event e_i occurring n times in the neighboring nodes at distance d and time offset t.

In [54], a mining algorithm was proposed by Chong et al. to discover strong rules from sensor data which were then applied for controlling the operation of the sensor network. They modified the Apriori technique to execute batch processing of the transactions in batches b_1, b_1, ..., b_n.

After collecting all sensory data, a rule in the form of a_n is generated which implies a_{n-1}, where n corresponds to the nth batch. The feature of this algorithm is that only a_n is sent to the base station, but using the knowledge of the rule a_{n-1} can be deduced. A repository stores all the rules that are extracted in this process. The proposed method was tested using a synthetic dataset.

Boukerche et al. have used sensor association rule to mine patterns from sensor nodes [66]. Unlike the works in [52], [53], Boukerche et al. used behavioral data that describe node activities to form the association rules. Their work explored the following two main approaches.

Direct reporting: each sensor sends its behavioral data to the sink in the form of notification messages on the detection of events within the current time slot, λ. The nodes may not participate in the formulation of rules, but report directly to the base station via sinks without storing data.

Distributed extraction: each sensor is equipped with additional memory to store behavioral data over a period of time, and thereby the computation and storage load is distributed over the entire network.

In [66], the process initiates with the distribution of the mining parameters, e.g, the time slot size, minimum support and the historical period, T_h among all sensor nodes throughout the network. Once the parameters are received by the nodes, each node allocates a local buffer B of size (T_h/λ) for storing data, and the entry corresponding to the time slot t_i is denoted as $B(t_i)$. As a sensor in the network continuously checks for an event, it sets the corresponding bit entry for the time slot during which it detects an event. This way the buffer becomes populated based on the detection of events by the sensor over the historical period. At the end of the period, a sensor counts the number of set bits in the buffer during that period, and if the count exceeds the given minimum support value, it sends a message or multiple messages to the sink. The message will contain the identifier of the sensor and the time slot numbers in which the buffer entry is set in response to event occurrences. Note that, in this approach, those sensors whose number of set bits is lower than the minimum support value does not play any role in forming the association rule. On receiving messages from all sensor nodes in the network, the sink places all nodes that reported an event occurrence at the identical time slot in the same epoch. The database then stores the epoch, which in fact then contains the sensor activities within that historical period.

Boukerche and Samarah in [67] further extended their work in [66] and proposed an in-network reduction mechanism that reduces the amount of the behavioral data that are extracted from a WSN. These data are required to generate sensor association rules. Experimental results show that in-network reduction gives better performance in terms of energy consumption and the number of messages needed to report the behavioral data compared to the direct and distributed extraction mechanisms of [56].

Existing most sensor association rules mining techniques for sensor networks require the behavioral data, which describes the sensor behavior, to be sent to the sink node by the sensors and then build a sensor database and apply different algorithms to find association rules from that sensor database. Retaining the computation at the node, in [68], an in-network mechanism is proposed to discover sensor patterns in the sensors themselves where sensors send only the frequent sensor patterns to the sink, not the sensor activity sets.

Wu *et al.* [69] proposed a decentralized technique to detect events, prune the irrelative events and find their temporal correlations. To mine event association rules they used MapReduce based technique called MapReduce-Apriori that uses the computational resource of multiple dedicated nodes of the system. Performance analysis shows that this technique attains nearly ideal speedup compared to centralized mining techniques. However, this technique not tested on a real dataset and it uses Apriori-based techniques that generate more candidate patterns.

Many centralized association rules mining techniques are proposed in the literature to discover meaningful patterns from sensor data streams. One problem faced in this case in the missing sensor data where may occur due to many reasons, for example, sudden disturbance in the communication link. An approach to estimate such mission values, termed Window Association Rule Mining (WARM), has been proposed by Gruenwald [55]. Using the proposed mining technique, WARM identifies those sensors that are related to the sensor whose reading is missing. Since sensor data readings are generated as a stream, it is not possible to apply Apriori [13] like association mining technique directly to the data stream. To adapt the Apriori algorithm for sensor stream data they proposed a framework called Data Stream Association Rule Mining (DSARM) in which several modifications are made to the Apriori scheme to adapt it for sensor streams. The modification can be listed as:

1) At first, rules are generated between pairs of sensors instead of generating all of the possible rules.
2) Then, the association between pairs of sensors is evaluated with respect to a particular state of the sensors, and this modification will lead to rules of the form $s_1 \Rightarrow s_2 = st$, which means that s_1 determines s_2 with respect to state st.
3) Finally, the sliding window technique is implemented to generate the association among sensors within the given window size.

In this framework, the readings of the related sensors in the current round participate in estimating the missing values. If a missing value cannot be estimated by using association rule mining, it is estimated using the average of all available readings for the sensor with the missing value.

To permit a fast estimation of the missed reading, a cube data structure is proposed to efficiently store sensor readings. Sensor data arrive in rounds. The cube is used to keep track of the state of individual sensors and the pairs of sensors that have the same reading. Figure 5 shows the proposed data cube. The horizontal and vertical dimensions of the cubes are

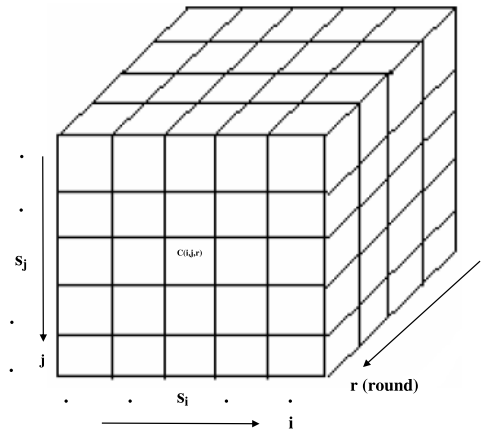


FIGURE 5. The cube model.

the sensor identifiers, and the depth dimension is the number of rounds (i.e. window size). $Cell(i, j, r) = st$ in the cube means that sensors s_i and s_j have the same state st at round r.

For around r and a missing reading from sensor s_n, the estimation process starts by deciding all the frequent states of sensor s_n (i.e., those states that have 'support' greater than, or equal to, the given minimum support). Then, for each frequent state st, all the possible rules of the form $(s_i \Rightarrow s_m/st)$ that meet the given minimum support and confidence, are generated. The weight of the contribution of each sensor, appearing in the antecedent of the frequent rules toward the missed reading with regard to state st, is then computed. This weight is based on the number of state match between sensor s_i and sensor s_m, within the given window size. The missed value is then estimated, based on the weight of each sensor.

In [70], [71], Jiang et al. propose a data estimated technique, called Closed Association Rule Mining (CARM), which can derive the most recent association rules among sensors based on the current closed itemsets in the sliding window. This method, based on the closed frequent itemsets mining algorithm in the data stream, is named CFI-stream [72]. It maintains an in-memory data structure, called direct update (DIU) to store closed itemsets. The extensive results presented in their works show that the algorithm achieves time and memory efficiency. However, the algorithm estimates the missing data according to the frequent patterns which are pre-computed based on the existing data. This raises a problem that, if the pattern containing the missing data does not appear in the frequent patterns, the missing data cannot be estimated. Figure 6 shows the DIU tree after receiving the first four transactions. It shows that currently there are four closed item-sets: I_3, I_1 I_2, I_3 I_4, and I_1 I_2 I_3 in the DIU tree, and their associated supports at the right upper corner are 3, 3, 1, and 2. A basic set of rules is generated from these frequent item-sets. All other rules can be inferred from this basic rule set.

In [56], Boukerche et al. propose a framework to mine the associations between the sensors in a particular sensor network. Their contribution can be summarized as

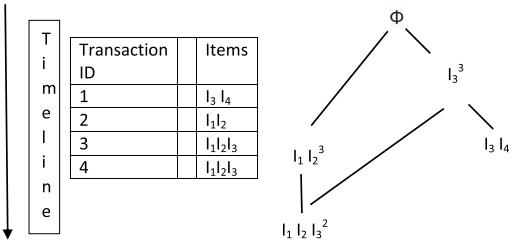


FIGURE 6. Lexicographical-ordered direct update tree.

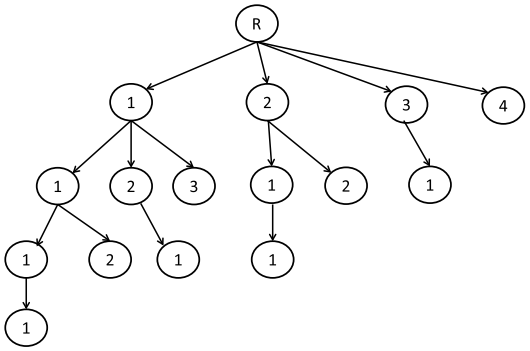


FIGURE 7. The PLT tree.

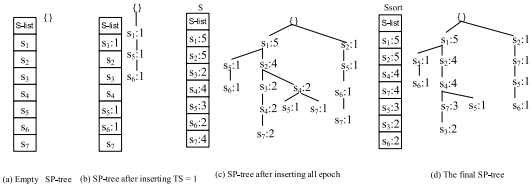


FIGURE 8. SP-tree construction.

follows: first, a new formulation for the association rule mining problem that makes it applicable to generate associations between sensors; second, Positional Lexicographic Tree (PLT), a new representation structure that can compress the sensor data to store in the database; third, a mining algorithm that can generate the frequent patterns from the PLT efficiently. The PLT Tree construction mechanism for the sensor set s_1, s_2, s_3, s_4 is shown in Figure 7.

In [58], a tree-based data structure called a sensor pattern tree (SP-tree) has been proposed by Tanbeer et al. With only one scan over the sensor database, an SP-tree generates a set of all association rules from sensor network data. The method calculates the frequency of event occurrences from the sensed data, build a prefix-tree using that information in any canonical order, and then reorganizes the tree in the descending order of frequency. The reorganization of the tree puts those nodes that frequently detect events at the top part of the SP-tree, thus resulting in a compact tree structure. To mine the set of frequent event-detecting sensors, the FP-growth mining is applied. The construction process of SP-tree is shown in Figure 8 for the SD presented in Table 2.

The performance of SP-tree was evaluated and compared with PLT [56] in terms of memory consumption and runtime.

The results reported that SP-tree performed better that PLT in both the aforementioned metrics. One reason for such better performance is that SP-tree construction requires only one database scan while PLT needs two database scans, clearly an advantage over time. The other reason is that the frequency-descending tree structure makes the mining of SP-tree more efficient than that of PLT.

In [73], Pan et al. introduced an improved version of the DSARM framework which addresses the drawback of [55] by taking consideration of the pairwise relationship between two sensors only; in this process, their relationships with other sensors are ignored. Neighbors' sensed data can reveal their spatial correlation dynamically. The improved framework exploits this spatial correlation to estimate missing sensor data using adaptive multiple regression.

In [74], Paik et al. proposed a technique to mine association rules from XML stream data. It involves a reformulation of the association rules for blockwise stream data and for the entire stream as well as making a list-based structure for storing XML tree labels. Woo *et al.* [75] proposed a technique of generating a new type of an association rule, called a context association rule, over an online sensor/actuator stream data and also introduced a prefix tree structure that captures all frequent context itemsets over the current data stream of sensor networks. Context association rules can invoke proper operations of actuators relevant to the values of the sensors.

To discover any correlation that exists among a set of targets, Samara et al. proposed a target-based association rule (TAR) technique in [76]. In their study, the targets were the locations of a missed reported event in a WSN deployed in a border area. In this technique, sensors need to use additional memory to store event data which incurs an increased cost of deployment.

Pal and Kumar in [77] proposed a distributed data collection model where an enhanced Apriori algorithm with MapReduce was utilized to find the frequent patterns from real and synthetic sensor data. They used the direct and indirect data collection mechanisms introduced in [94] to collect the metadata from IoT applications and store these data in the database based on timestamp. Then the MapReduce framework is used to mine frequent patterns from the database by using a distributed Apriori approach. However, the experimental results were not extensively analysed for the proposed method. In [83], an improved version of frequent pattern mining technique has been proposed that targeted IoT big data. They modified the Apriori algorithm as an advanced Apriori algorithm (ADAA) to identify an item's association where the ADAA is less sensitive to the minimum support threshold compared to the traditional Apriori algorithm. However, the proposed method is not suitable to deal with the scalability issues of IoT.

The proliferation of smart home applications and remote monitoring has generated a number of interesting behavioral pattern mining applications. Nazerfard [80] proposed a model to mine the temporal aspects of activity patterns to help the living setting in a smart home scenario by using temporal features to determine the temporal correlation of different activities with the begin time and duration features. This type of activity patterns can also be utilised in home automation, context-aware activity reminder systems, and anomaly detection in smart environments. They used the frequent pattern mining technique to mine activity patterns. The work, however, did not explore how the model can be extended to IoT based smart home applications. Lee et al. in [84] proposed a privacy detection scheme called PDS to assess privacy from IoT based smart home applications where they used mining results to identify elders' movement activities by analysing sensor data and map global sensor topology from these activities. They identified locations for deploying sensors based on the association rule mine techniques. However, the work lacks extensive performance analysis. In [82], Kireev et al. proposed an association rule based predictive algorithm to predict the need for repair of different units (e.g., heating, air conditioning) in IoT based smart homes. The proposed model has two steps. Firstly, the sequences of signals from sensors are extracted by using the association rules mining technique and secondly, a classifier is used to identify the generated patterns to group them. This approach is yet to be adequately evaluated through experimentation.

To mine temporal patterns over the clinical sensor data stream and apply that knowledge for patient welfare, Banaee and Loutfi [78] proposed a data-driven rule mining technique to mine temporal patterns over the clinical sensor data stream. The proposed method is divided into three phases: temporal rules mining algorithm, temporal rule set similarity calculation and temporal rule representation. At first, the rules mining algorithm performs prototypical pattern abstraction and then applies temporal rule mining. A similarity measurement technique is used to compare the generated rule sets among themselves, and then based on the extracted rules it is identified how a clinical condition is distinct from others. Finally, a natural language generation technique is used to represent these generated rules into text which are able to provide physicians with significant information to guide a patient's treatment plan. Though promising, the semantic model used in this work needs to be improved to represent the temporal rules into text.

Rani and Pushpalatha [81] proposed a sliding window-based vertical partitioning parallel and distributed algorithm (VPPDA) that utilizes a MapReduce paradigm to discover FPs from sensor data and can be used for remote monitoring applications. In the model, each window is assigned to different nodes and each node runs the VPPDA algorithm in parallel to generate FPs. VPPDA removes the overhead of inter-process communication in the overlapped window concept which ensures better performance. Since this model does not remove the old information from the window, it may accumulate garbage information as the time progresses.

In [85], an incremental processing method has been proposed for frequent subgraph detection where they modified the data stream tree (DSTree) [97] and used a DSMatrix

TABLE 3. Comparison of Association rules based techniques for WSN/IoT.

Author	Year	Method	Association between	Application	Data source	Strengths	Limitations
M. Halatchev et al. [55]	2005	Apriori like Centralized extraction	sensors	Traffic Monitoring	Synthetic	Estimate missing values	Ignore sensor that report multiple time
Loo et al. [52]	2005	Loosy counting	Sensor values	WSNs monitoring	Synthetic	Compute the exact set of rules	Time consuming
Romer et al. [53]	2006	Apriori like	Sensor values	Environmental monitoring	Real, synthetic	Estimed missing values	Communication overhead
Chong et al. [54]	2008	Apriori like	Sensor values	WSNs monitoring	synthetic	Use for energy savings	Takes much time
Boukerche [66] et al.	2007	Positional Lexicographic Tree (PLT)	Sensors	Monitor WSNs quality of service	Real, synthetic	Predict source of future events	Increase cost due multiple
Gruenwald et al. [70]	2007	Freshness Association Rule Mining (FARM)	Sensor values	Environmental monitoring	synthetic	Estimation of missing sensor	Unable to handle high speed stream data
Jiang et al. [70]	2007	FP-growth	Sensor values	Data Analysis	synthetic	Compute exact set of patterns	Inefficient for handling high-speed data
Tanbeer et al. [58]	2009	SP-tree	sensors	Generic monitoring	synthetic	Discover events patterns	High tree build cost
Boukerche et al. [56]	2009	Minimum node data gathering ree (MNDGT)	sensors	Area Monitoring amount of data	synthetic	reduce the candidates	Time consuming
Samarah et al. [76]	2009	Target	TAR	Boarder monitoring	synthetic	Predict the source of the future events	Need extra storage chip for each sensor
Boukerche et al. [67]	2009	In-network	Sensors	WSN performance	Synthetic	Less message generate	not tested on real environment
Anjan Das [68]	2012	In-network mechanism	sensors	Generic monitoring	synthetic	Estimation of missing sensor values	put extra load on the sensor node
Wu et al. [69]	2013	MapReduce -Apriori	Sensor value	Networking monitoring	Synthetic	useful in handing large data	Generate huge candidates

to store data. Finally, they used a sliding window-based technique to mine recent frequent subgraphs from the pattern growth approach. However, this method did not explore the real IoT application and its performance may not be acceptable in big data scenarios.

Behavioral pattern mining has also been explored for estimating accuracy of query execution. In [79], a geometric query intersection problem has been explored where top-k patterns with a temporal granularity are extracted rather all frequent patterns to save the memory. The proposed method starts with data pre-processing where data cleaning is performed by using null and missing values. In the query, the processing module deals with user query that is based on spatial results. In the calculation of the relevant results module, the matched results are collectively stored and then top-k results are generated by using the query algorithm. Lastly, the accuracy of the query is evaluated. The main limitation of this mechanism is it does not provide any information on how to collect the multivariate temporal data. A comparison of association rules based BP techniques for WSNs and IoT is shown in Table 3 & 4.

C. INTERESTINGNESS BASED BPS MINING TECHNIQUES FROM WSN/IOT DATA

The previous section presented works where the generation of sensor association rules is based on the frequency of occurrence of patterns. Often these techniques produce a large number of rules, most of which may bear little significance to the user/application or are unable to capture the actual correlations among the sensor data. As a result, another type of behavioral pattern called *associated sensor patterns* is proposed by Rashid et al. in [43], [63], [98] to improve on these aspects. This type of behavioral pattern captures association-like co-occurrences as well as temporal correlations which are linked with co-occurrences. These works capture patterns in a compact tree structure, called associated sensor pattern tree (ASP-tree) that utilizes a pattern growth-based approach to generate all associated patterns with only one scan over the dataset. An ASP-tree is then mined using a mining algorithm (ASP). ASP-tree has a few similitudes with a SP-tree [58] in the construction and reconstruction mechanism. However, an ASP-tree performs an additional compression technique like the Patricia tree [99] and Cantries [100]. This ensures

TABLE 4. Comparison of Association rules based techniques for WSN/IoT, continued.

Author	Year	Method	Association between	Application	Data source	Strengths	Limitations
Pan et al. [73]	2014	Multiple Regression Algorithm	Sensor values	Data Analysis	Real	Estimates missing sensor values	Not efficient for large WSNs
Paik et al. [74]	2014	FP-Growth	Sensor value	Generic monitoring	Not specified	Generates redundant rules	No experiment is done
Woo et al. [75]	2014	CAR-tree	Sensor value	Generic monitoring	Synthetic	Captures frequent contexts of a user	Memory inefficient
Pal et al. [77]	2017	Apriori MapReduce	Sensor	Network monitoring	Synthetic, real	Highly scalable	Generates many result sets
H Banaee et al. [78]	2015	Similarity method	Sensor value	Clinical monitoring	Real	Checks similarity among rules	Inefficient for Big sensor data
Raman et al. [79]	2017	FP-growth	Sensor value	Generic Monitoring	Real	Scalable	Details missing about sliding window
Ehsan et al. [80]	2018	FP-growth	Sensor value	Smart home	synthetic	Not generate redundant patterns	Not suitable for Big data
Rani et al. [81]	2019	MapReduce	Sensor value	Remote monitoring	synthetic	Highly scalable	Not tested on real data
Kireev et al. [82]	2018	FP-growth	Sensor value	Smart City	Real	Effectively predicts sensor state	Dataset details is missing
Wang et al. [83]	2019	Apriori	Sensor value	Generic IoT	Real	Improves operational efficiency	Generates many candidate patterns
Lee et al. [84]	2019	Association rules	Sensor value	Smart home	Real	Finds privacy information	Inadequate performance analysis
Bok et al. [85]	2018	DSMatrix	Sensor value	Generic IoT	Real	Reduces duplicate operation	Inefficient for Big data

ASP-tree holds less number of nodes than SP-tree due to residing the same support sensors in a single node. Subsequently, the ASP-tee memory footprint could be far less than the SP-tree. Besides, SP-tree utilizes the FP-growth based mining method to generate frequent patterns. FP-growth mining is not directly applied to ASP-tree because an ASP-tree not only mines the frequent sensor patterns but also frequent associated sensor patterns. Therefore, a pattern growth mining technique is devised that can handle the additional feature of the ASP-tree. Table 6, shows the memory comparison among ASP-tree, SP-tree, PLT and FP-tree for T10I4D100K data [101] and Intel data [102].

In a dynamic environment, recent data from sensors bear higher significance. To accommodate this issue, the ASP-tree construction is again improved by running a sliding window over the data with time and updating the tree structure as the window moves. This tree is called sliding window ASP-tree (SWASP-tree). Both trees have the 'build once and mine many' property, making it highly suitable for interactive mining. The step-by-step construction process of the SWASP-tree for window 1 and window 2 based on the sensor data stream (SDS) of Figure 3 is shown in Figure 9. The pseudo-code of SWASP is shown in Algorithm 1.

The proposed SWASP-tree appears to be similar to the data stream (DSTree) [97] and compact pattern steam tree (CPS-tree) [103]; however, there are a few contrasts around these trees. In spite of the fact that the construction process of SWASP-tree is practically the same as DS-tree, there is no restructure-compression phase in DSTree. An alternate essential divergence between SWASP-tree and DS-tree could be seen throughout the tree update stage when the window slides. As stated by [97], throughout the tree update stage, the DSTree does not traverse all nodes and does not shift the frequency count list at every node. Consequently, the frequency list is not updated for those nodes which are not visited during the new incoming batch. Such updating is likely to create some invalid nodes in the DSTree structure for the present window. This may result in a tree that is over-burdened with additional nodes carrying insignificant information for the current window. The DSTree for window 1 and window 2 of our example SDS is shown in Figure 10.

Because of the possible additional nodes, the DSTree must use extra computational overhead throughout the mining process to erase or to give careful consideration to overlook them. Conversely, SWASP-tree performs the frequency shift operation for every node and it doesn't hold any additional node that will cause a burden for the present window.

The structure of SWASP-tree is very unique in relation to the CPS-tree. CPS-tree contraction starts with the insertion of all the items in the lexicographic item order and there are two

TABLE 5. Comparison of Interestingness measure based techniques for WSN/IoT.

Author	Year	Method	Association between	Application	Data source	Strengths	Limitations
Ismail et al. [86]	2019	TMCP-tree	Sensors	Body sensor networks	Synthetic	Discovers interesting patterns related to human daily life	No comparison with existing techniques provided
Ismail et al. [87]	2018	PPSD-tree and MapReduce mechanism	Sensors	Data Analysis	Synthetic	Computes productive periodic-frequent sensor patterns	PPSD-Tree construction may takes longer time
Bhuiyan et al. [88]	2016	DP-tree	Sensor values	IoT monitoring	Real	Event detection in low energy consumption	Inter node communication is very high
Rashid et al. [89]	2017	PASP-tree	Sensors	Network monitoring	Synthetic and real	Contains interval of ASP	Inefficient for big IoT data
krishna et al. [90]	2016	Dissimilarity function	Sensor values	Smart home	–	Prunes unnecessary patterns	No experiment provided
Harada et al. [91]	2019	MISCELA	Sensor values	Urban Management	Real	Useful in CAP mining	Does not consider big IoT data
Tianrui et al. [92]	2018	MapReduce mechanism	Sensors	Generic	Synthetic reduction	Runtime on real environment	Not tested on
Rashid et al. [93]	2013	RSP-tree	Sensors	Network monitoring	Synthetic	Captures temporal regularity	Inefficient for big IoT data
Rashid et al. [43]	2015	ASP-tree	Sensors	Monitor WSNs	Synthetic, real	Captures true correlation among sensors	Tree construction and reconstruction takes longer time
Rashid et al. [61]	2015	ShrFSP-tree	Sensors	Monitor WSNs	Synthetic, real	Captures share relation among sensors	Does not consider multiple thresholds
Rashid et al. [94]	2017	MapReduce	Sensors	Monitoring WSNs	Synthetic, real	Captures temporal regularity	Increased number of result sets increase
Yassine et al. [95]	2017	FP-growth	Sensors	Smart home	Synthetic, real	Captures irregular activity	Inefficient for big IoT data
Usman et al. [96]	2017	MMFP	Sensors	Generic IoT	Real	Generates fewer candidates	No real IoT applications considered

TABLE 6. Comparison among various mining tree structures in terms of memory usage.

Dataset min_sup(shown as $\delta\%$)	Tree	min_all_conf (%)	Memory (MB) δ_1	δ_2	δ_3
T10I4D100K $\delta_1 = 1.0, \delta_2 = 2.0, \delta_3 = 3.0$	ASP-tree	20	6.10	3.10	0.21
		40	5.50	2.30	0.15
		60	4.60	1.20	0.09
	SP-tree	–	7.10	3.20	0.31
	FP-tree	–	8.35	4.66	0.50
	PLT	–	7.50	3.60	0.35
Intel data $\delta_1 = 30, \delta_2 = 40, \delta_3 = 50$	ASP-tree	20	1410	480	250
		40	1250	390	170
		60	1090	320	105
	SP-tree	–	1470	495	270
	FP-tree	–	3800	1250	700
	PLT	–	1500	510	280

kinds of nodes accessible in each path of the tree: ordinary node and tail node (the last node of each transaction is called a tail node). Every node expressly holds the item id and the total frequency count for the item in their respective paths, whereas the tail node furthermore upholds the batch counter. The structure comparison of SWASP-tree and CPS-tree is shown in Figure 11.

SWASP-tree utilizes the same restructuring process like CPS-tree. In addition, SWASP-tree performs an additional compression technique like the one in [99]. This guarantee SWASP-tree catches less memory than DSTree and CPS-tree. The objective of both DSTree and CPS-tree is to mine exact recent frequent patterns from the current information stream where both trees utilize FP-growth like mining system. Then again, the target of SWASP-tree is to dig recent associated patterns from the current stream information by using a new pattern growth technique.

One major requirement of a sliding window based method is the selection of window size and determination of appropriate size requires sufficient knowledge about the nature

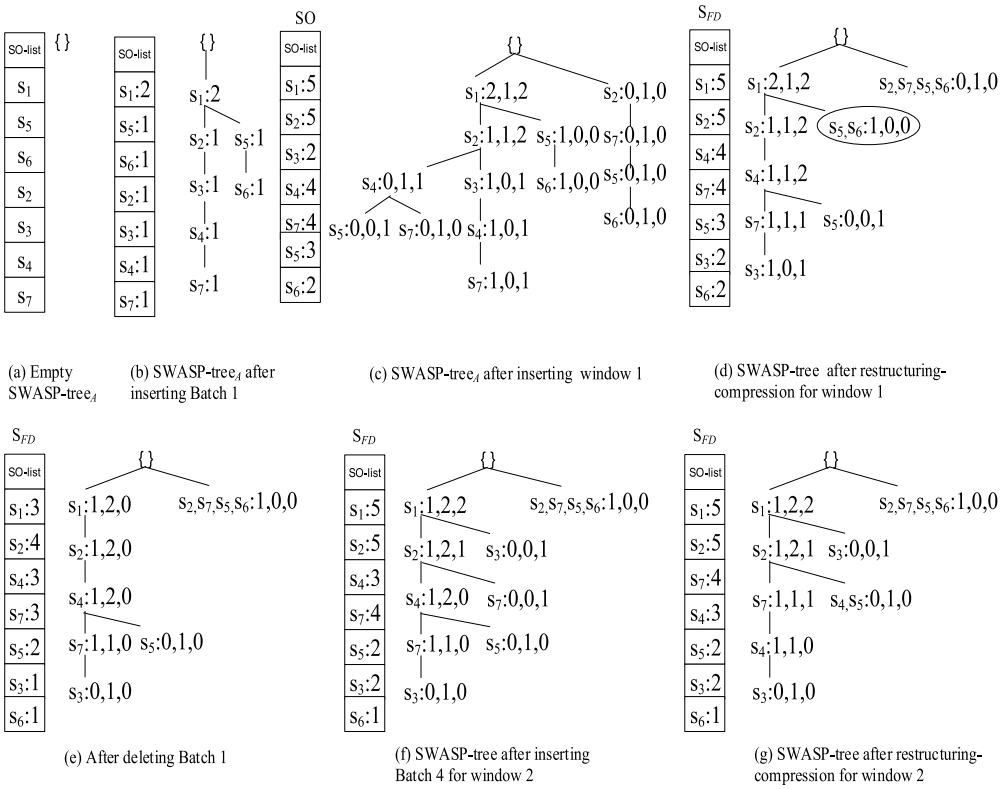


FIGURE 9. Construction of SWASP-tree (window 1 & window 2).

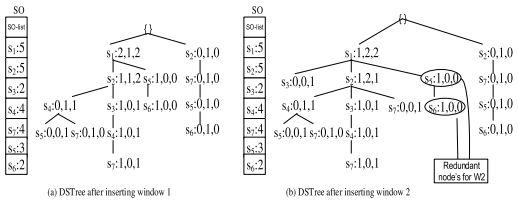


FIGURE 10. Construction of DS-tree for two windows.

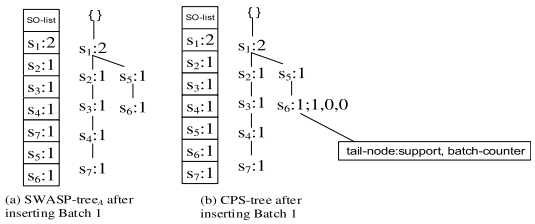


FIGURE 11. Structure comparison: SWASP-tree v/s CPS-tree.

of the data stream. However, gaining sufficient knowledge about the data stream is challenging for various reasons, e.g., unpredictable change in the data stream or operational condition of the application. To address the issue of window size selection, a slightly modified version of SWASP, called Adaptive SWASP (ASWASP), is proposed. ASWASP dynamically adjusts the window size taking computational resources into consideration [104]. This varied window size, in turn, varies the rate of processing of data streams and handles unpredictable changes in the nature of the data. The method works on defining three parameters, namely, stream rate (*sr*), memory usage (*mu*), and overall run time (*rt*) and formulating a utilization factor in a similar way as in [105]. The rate of arrival of sensor data is defined as stream rate, the average memory used by the sliding window epochs is the memory usage, and the average runtime of windows during an observation span is called runtime. The utilization factor, *UF*, is formulated as

$$UF = [(rt(second) * sr) + sin(\pi/180 * mu)] \tag{3}$$

In the above equation, the purpose of the sine function is to limit the value of UF between 0 to 1. Depending on the value of UF, the sliding window size is incremented or decremented. A threshold is first set for UF, and if the currently observed UF value is lower/higher than that threshold, the window size is then reduced/increased. Such adjustment of window size will make better use of memory to cover maximum epochs.

The above works by Rashid *et al.* [43], [63], [98] considered only the binary frequency of a pattern. However, consideration of the binary frequency of a pattern is not a sufficient indicator for finding meaningful patterns, rather trigger value should be used. Rashid et al. in [61], [64] proposed a new type of behavioral patterns called share-frequent sensor patterns (SFSPs) by considering the non-binary frequency values of sensors in epochs. SFSPs can find

Algorithm 1 SWASP Algorithm

Input: *SDS*, *min_sup*, *min_all_conf*, current window W_c, no. of batches in W_c, no. of epochs in a batch (M), *Initial_Sort_Order*
Output: ASP for W_c

Begin
$SO \leftarrow$ an SO-list arranged in *ISO*
SWASP-tree$_A \leftarrow$ a prefix-tree with null initialization
for each batch B_i **do**
  if j > N then **then**
    Call DeleteSWASPtree(R)
  end if
  for each epoch E_k in batch B_i **do**
    Sort the items of E_k in the order of SO
    Update support value in the SO-list
    Call InsertSWASPtree(E_k, I, N, R)
  end for
  Calculate S_{FD} from SO in frequency-descending order using merge-sort method;
  for each branch in SWASP-tree$_A$ **do**
    Sort the branch in S_{FD} using branch sorting method (BSM);
  end for
  for each branch in restructured SWASP-tree$_A$ **do**
    Identify the *same support sensor node* in each branch and merge them to a single node
  end for
  while any mining request from the user **do**
    Input *min_sup* and *min_all_conf* from user
    for each sensor υ from bottom of SO-list **do**
      Call Mining (CPB_υ, SO-list$_\upsilon$, υ)
    end for
  end while
end for
End

a correlation among a set of sensors and hence can improve the performance of WSNs in a resource management process. A share-frequent sensor pattern tree (ShrFSP-tree) has been proposed to facilitate a pattern growth mining technique to discover SFSPs from WSN data.

To process a large amount of data from the WSNs, a parallel and distributed framework that uses a multi-knapsack optimization formulation for efficient balancing of load and memory usage among processing nodes is also developed which is termed as parallel ShrFSP-tree (PShrFSP-tree). This framework significantly reduces the I/O cost by capturing the local database contents with a single scan and uses a fully parallel pattern growth mining technique with reduced interprocessor communication overhead, and therefore is highly scalable. Results demonstrate that ShrFSP-tree outperforms its counterpart [106] in static databases in terms of execution time by 35% and memory usage by 55%.

Figure 12 illustrates how the PShrFSP-tree performs in the homogeneous and heterogeneous environments for two widely used datasets, namely, the T10I4D100K and Intel data. From Figure 12 it is evident that the runtime of the PShrFSP-tree in the heterogeneous environment is substantially lower than that of the homogeneous system. The higher performance by the heterogeneous system is achieved through a better distribution of load among the nodes using the knapsack problem-based load balancing technique. Consequently, the computational time required by each node is nearly equal and the overall computational time decreases.

Temporal regularity of a pattern (i.e., whether a pattern occurs at regular intervals, or in irregular fashion) can be an important measure for many applications. It will be very significant to develop a temporal regularity based sensor data mining model. To this end, Rashid *et al.* [62], [93] have proposed regularly frequent sensor patterns (RFSPs) which are based on temporal regularity of behavioral patterns. RFSPs can discover a set of sensors that are temporally correlated and appear at regular intervals, revealing important knowledge from the collected sensor data. Since distributed schemes offer better reliability and availability, a distributed data acquisition model [94] is proposed for collecting data from sensor networks that are required for mining RFSPs. The advantages of RFSP are that it requires less memory and attains a compact tree structure, and its construction requires only a single database scan. All these make the RFSP mining technique highly attractive with low runtime.

Since MapReduce becomes a defacto model for big data analysis, a parallel implementation of RFSP on the MapReduce platform, called RFSP on Hadoop (RFSP-H) [94], is proposed which can mine a large scale sensor data with further efficiency. Figure 13 shows the proposed model for RFSPs mining on the MapReduce platform which is capable of handling large scale sensor data mining in the IoT scenario. The runtime comparison of RFSP-H with a single-processor based RFSP-tree on Intel data [102], by varying the *max_var* values where *min_sup* was fixed at 20%, is shown in Fig. 14. Experiments with different *min_sup* values show a similar trend. These results show that, for mining data from large IoT networks, Hadoop offers a promising and reliable platform for efficient mining with the reduced response time.

The same authors in another work [89] proposed a new type of behavioral pattern called periodic associated sensor patterns (PASP) where a compact tree structure was used to mine these patterns from WSN data. Experimental results demonstrate that the proposed technique effectively determines the PSAPs from large WSN data.

In [87], Ismail et al. proposed productive periodic-frequent patterns from IoT data. The patterns are called 'productive periodic' as they are the periodic patterns that exit because of some predefined associations. Such patterns are useful for human analysis like disease vulnerability where an individual may often succumb to certain seasonal attacks or fashion choices where some people have preferences for certain colors or styles. To discover these patterns efficiently from large scale IoT Healthcare data, they used a MapReduce-based parallel mining technique. Implemented over the Hadoop

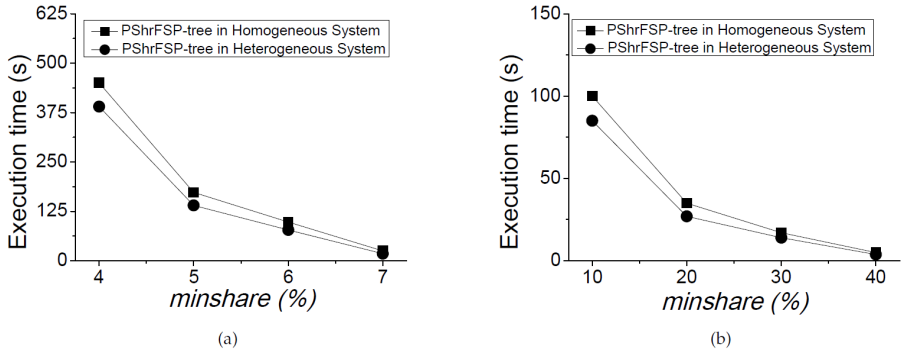


FIGURE 12. Comparison of execution time in homo and heterogeneous systems on PShrFSP-tree: (a) *T10I4D100K* and (b) *10 Days data*.

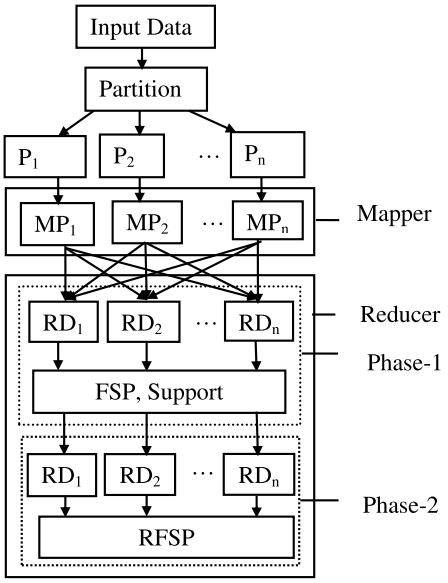


FIGURE 13. Regularly frequent sensor patterns mining for sensor data on MapReduce platform.

platform, experimental results show considerable execution time efficiency of this approach. In [92], Tianrui et al. proposed a MapReduce based framework to mine representative patterns in the IoT environment. Their proposed technique can mine the represented patterns in a reasonable time. Though parallelism helps to reduce runtime, the total number of result sets is increased. Two optimization strategies were also proposed to further improve runtime efficiency. None of these works, however, did not suggest how load can be balanced among the processors to avoid delay in receiving intermediate mining results from an overburdened processor.

Ismail et al. in another recent work [86] proposed a technique to mine regular human activities through discovering patterns over a non-uniform temporal database created from the data collected from body area sensor networks. Here the main challenges were how to deal with different periodicity and supports values of a pattern and find the correlations among the identified patterns. Another aspect of this work is to consider contextual variation among patient activities in mining. Mining such patterns can help to design a personalised treatment plan for patients, specially for patients in aged-care facilities. Since context depends on the environment, situation or action (e.g., a patient's behavior may change with the lighting or temperature condition), defining context attributes properly and comprehensively still remains a challenge. Though the work in [86] constructs a tree structure to capture context variation, its efficacy to handle such variation has not been adequately evaluated through experiments. Moreover, a semantically enhanced data from smart wearable and environment sensors might provide the opportunity for further advancement in this area.

In [88], [107], Bhuiyan and Wu proposed a data mining model to mine differential sensor patterns (DSP) for the IoT environment. The proposed model considers a parallel and distributed scenario and discovers a pattern of sensors which contain the event information. To generate DSP they used DP-tree. The DSP mining technique was shown to extract event indicators where TAR [76] and MAR [56] mining failed to detect. One aim of the work was to reduce the energy spent to support the additional message exchanges among sensors needed during the data preparation phase of the mining task. Their computation of energy cost is based on equal-sized cluster of sensor nodes, however, in practical deployment clusers often are of unequal size.

A temporal pattern mining technique that uses a fuzzy similarity measure to reduce the candidate patterns by pruning based on estimated support bounds has been proposed in [90]. This approach takes less computation time since it prunes the patterns at an early stage before the actual calculation of support values. The main contribution of this work is the fuzzy similarity measure and the pruning rules. However, the authors did not present any experimental works at all to verify the claimed computational efficiency. Yang in [108] proposed a conceptual model to link group ranking problem, and thereby stores group-ranking sequences in a database in addition to the sensor database. Pattern discovery requires the use of both databases. The mining algorithm requires the sensor data to be transformed to ranking sequences before being

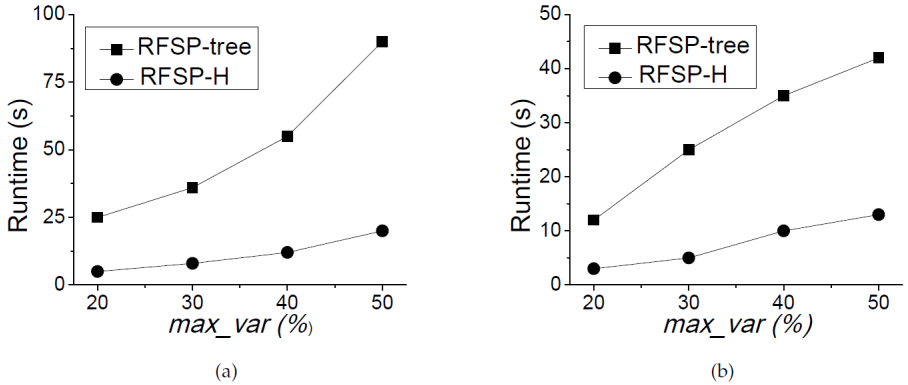


FIGURE 14. Comparision of execution time in RFSP-H and RFSP tree, (a) *S300H10dT60s data* and (b) *S150H10dT60s data*.

used. Though the work is useful to extract frequent sensor ranking patterns from sensor data, it was solely a conceptual work without being tested on any simulated or real-world data and hence its effectiveness is yet to be assessed.

K Harada et al. in [91] proposed correlated attribute pattern (CAP) mining technique from sensor dataset. To discover CAP they proposed a technique called MISCELA which can reduce the computational cost. They also proposed a CAP search tree structure to store data and then applied the mining technique on it. CAP discovers patterns that are spatially co-related and close and likely to co-evolve in time, and therefore, it is useful in smart city applications like road traffic and congestion monitoring in big cities. The drawback of the technique is that the response time to a CAP search can be very large when the number of sensors and CAP attributes and the evolving rate of data become large. This makes it unsuitable for large scale WSN or IoT data.

In [109], Cheng et al. proposed a robust and structured event-driven, application-oriented platform architecture for IoT network management that combines the merits of service-oriented architecture (SOA) and event-driven architecture (EDA). They also introduced a situational event pattern and a situational event-driven service coordination behavior framework that led to a detection algorithm capable of identifying mismatch in the ruleset. Yassine *et al.* [95] proposed a framework that uses IoT based smart home application data to discover human activity patterns. The proposed method consists of the following four steps: data preparation, extraction of frequent patterns, incremental analysis using clustering, and activity prediction using Bayesian networks. To find irregular human activities, they used frequent pattern mining integrated with cluster analysis which ultimately identifies people having difficulties in taking care of themselves (e.g., not taking bath or making food, forgetting medicine). Later the Bayesian network was used to build the activity prediction model. The work lacks a proper comparison of the proposed model with other similar methods.

In [96], Usman et al. proposed a model to mine malicious frequent patterns (NMFPs) from IoT environments which can be utilized to detect anomalies. The proposed model consists of three phases: feature selection, legend creations, and frequent pattern mining, and has low computational costs. The authors compared their model with the existing association rule mining techniques such as Apriori, FP-Growth, and Prefixspan [110]. The proposed MMFP algorithm works on two user-defined thresholds named minimum support value (MSV) and confidence value (CV). After receiving the input data, the MMFP algorithm mines frequent patterns based on MSV and CV with only one scan. Then an expert analysis is performed to extract malicious patterns. These methods need to be optimized for the real-time data distribution service for IoT applications.

A comparison of interestingness measure based BPs techniques is shown in Table 5.

D. AN BEHAVIORAL PATTERN MINING FRAMEWORK FOR IOT

The existing association rules based behavioral pattern mining techniques have many limitations. Sensor association rules suffer from the lack of appropriate methods for the selection of appropriate support and confidence values. Existing techniques [52], [53], [55]–[58] only consider the homogenous data. However, for real-time decisions, homogeneous data may not be appropriate. Most techniques utilize the centralized method [43], [56], [58], [61], [62], [64], [93] where the data are sent to a central node (i.e., sink) for pattern discovery. Since data need to travel from the source to the sink, such methods suffer from overhead and long response time due to this communication because the distance might be long depending on the network size. Although the response time and the energy consumption can be improved by adopting distributed methods discussed in the preceding section, they will suffer from similar shortcomings if the cluster heads need to control a huge number of nodes. Moreover, existing methods consider only temporal, spatial, or spatiotemporal correlations [53] among data and fails to take account of the attribute dependency among sensors.

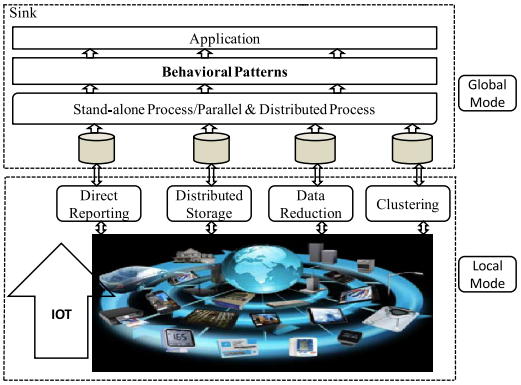


FIGURE 15. A proposed framework for behavioral pattern mining in IoT.

The lack of consideration of attribute dependency enhances the computational cost and reduces mining accuracy. Furthermore, existing techniques (e.g., [86], [87], [90], [92]) are not evaluated by real deployment and not tested or suitable for IoT big data (e.g., [63], [95], [96]). To resolve these shortcomings, here we propose a knowledge-based framework that can mine the pattern in online as well as offline which is shown in Figure 15. For example, if we consider this framework for an IoT based smart city application scenario (e.g., parking area monitoring), initially the sensor nodes perform mining tasks locally and then send the processed data to the sink, if it is required, and make a global view of the whole network. Based on this data, the IoT application can serve an end user query.

The proposed model has two parts: the local mode and Global mode. In the local mode, the sensor data coming from the IoT are collected by using direct reporting, distributed storage, data reduction or clustering mechanisms. Depending upon the application purpose different data collection models are used. In the direct data collection model, the data are directly stored at the data storage location. However, for the distributed and data reduction mechanism sensor nodes utilize their processing capabilities to perform mining operations locally and only forward partially processed and required data to the storage location.

In the global mode, the data are processed by using parallel and distributed models to discover desired behavioral patterns. Since IoT generated a huge amount of data, and these data need to be processed within a short time so parallel distributed models are very necessary. Finally, based on the given knowledge formulation we discover behavioral patterns from these data.

VII. OPEN RESEARCH ISSUES

Although different methods have been proposed in the literature to mine behavioral patterns from WSN data there still remain many research issues in this area which are outlined below:

1) Concept drift in data stream:
In data mining, unexpected changes in the underlying data distribution over time are known as concept drift [111]. The changes in underlying data can happen due to changing personal interests, changes in population, or adverse activities. In relation to IoT, changes can be attributed to a complex nature of the environment, the scale of activity patterns in the monitored area, the occurrence of emergency events, etc. [112]. Existing FP mining techniques consider that the data distribution characteristics in the sensor data stream will not experience any drastic change and thereby, ignore the presence of any concept drift in the data. However, in reality, concept drift is very likely to occur in sensor data stream [113], and if not taken into consideration, the mining results will be inaccurate. Therefore, it is essential to develop an intelligence model that can adapt to this dynamic changing nature of data accurately in real time and this can be an important direction of research in behavioral pattern mining of IoT sensor data.
In literature, several techniques are available to detect concept drift and identify the point of change or time interval during which changes occur [114]. Once the amount of drift exceeds certain threshold, an already built model can be rebuilt by adjusting the minimum support threshold depending on that amount. Gradual forgetting [115] and incremental learning [116], [117] approaches used in machine learning can also be explored here.
2) Context: Context is a very important issue in Big data. For transforming the raw data into real information, contextualization is crucial which not only reduces the size of the data but also helps to discover significant knowledge in a timely manner. This information can be utilized as a practical perception that allows intelligent corporate decision-making [118]. In general, the target is to find relevant and useful information that will enhance our intelligence. The European Union has recognized context awareness as a vital research area for IoT [119]. In case of IoT, a sensor's attributes like location, capability, time, velocity can be regarded as context items. Many studies have demonstrates the importance of context for IoT, such as context-aware trust model for lightweight IoT devices [120]. Consequently, it is an essential factor to consider the context to design a knowledge-based behavioral pattern mining technique.
3) Promote green IoT:
Attaining energy efficiency in any engineering system and low energy communications have become major research issues in recent times as they contribute to environmental benefits [121]–[123]. Green IoT (G-IoT) targets to implement environmental-friendly and energy-efficient features by adopting strategies at both hardware and software levels, and can be beneficial in reducing energy consumption from IoT-based applications [124]. Even small savings in energy in the operation of sensors and sensor networks will result in

large energy conservation because of the sheer number of sensors to be deployed by IoT services in the near future. In addition, replenishing energy in sensors is not easy in many sensor networks because of their inaccessible locations, and saving energy in such cases will increase their lifetime.

How knowledge gained from WSN/IoT data mining can be used in the energy-efficient operation of networks needs to be investigated to achieve G-IoT [125]. In this case, machine learning based energy optimization techniques can be useful to predict the energy consumption by identifying different activities in the IoT applications [126]. Moreover, using distributed control, fog computing and mass transit gains can ensure dynamic and distributed energy control model as well as low energy consumption [127].

4) Increase the efficiency of Actuation: The operational efficiency of standalone WSNs or multiple distributed WSNs in IoT will bring a number of highly valued benefits. These include early detection of events like bush-fires or nuclear leakage in a power station, actuation of actor network(s) in response to a detected event, and better coordination among WSNs for a coordinated task, like rescue operations after an accident or natural disaster. Little work has been done to date on how knowledge extracted from mining WSN data can be used in increasing the efficiency of wireless actor networks, and this remains an important research challenge.

   One way of faster detection of emergency services like the ones mentioned above is to perform data mining close to the event source, such as by edge devices in the IoT. A distributed and efficient mining algorithm can be developed which can run in the resource-limited environment in the edge devices, and a co-ordination and decision fusion scheme can be devised that can fuse mining results from multiple edge devices for increased detection accuracy of events spanning over large spatial area. Using the mining results as part of a query, an optimization problem can be formulated to find and then activate the actuators in actor networks that are in the best locations to mitigate the event.

5) Data security:

   Security in the big data world is likely to present a unique issue. Big data collected from IoT poses a number of security risks [128]–[130]. According to Analyst Gartner [131], "*The IoT, which excludes PCs, tablets and smartphones, will grow to 26 billion units installed in 2020 representing an almost 30-fold increase from 0.9 billion in 2009*". Big data generated from the IoT presents an absolutely new complexity when it comes to the need of security and privacy. Every single sensor or device that is connected to the internet represents a potential risk and one weak link could open up access to thousands of sensors or devices on a network with potentially serious consequences. Another crucial issue to protect the confidentiality of the data gathered through these sensors. In 2014, researchers from Eurecom found 38 vulnerabilities which include weak encryption and backdoor accessibility across 123 products of IoT devices [132].

   In practice, when data are collected by connected devices in IoT, one of the requirements is to provide high privacy assurance. The implementation of the 'defense-in-depth' mechanism proposed by the Federal Trade Commission (FTC) can reduce cybersecurity threats [133]. Moreover, deep learning-based techniques can be used for detection, modeling, monitoring, analysis and defense against various attacks to security systems [134]. Such techniques and IoT system vulnerability analysis [135] can be used in conjunction with frequent pattern mining to detect anomalies and intrusion as well as to strengthen security.

VIII. CONCLUSION

The emergence of IoT, with WSNs as its building blocks, has created a collaborative platform for better collection, distribution and management of sensor data, which can be static or stream data. IoT applications, ranging from smart city to undersea monitoring, are expanding to many other innovative applications with the fast deployment of networking infrastructures offering IoT services over long range wide area networks. This huge amount of sensor data from IoT will only be valuable if they can be mined for knowledge in real time, however, this presents many new challenges for the knowledge discovery techniques. This paper presented an overview of behavioral patterns mining techniques from sensor data in IoT, analysing their strengths and limitations. It provides a very good foundation for the researchers who are intersected to gain an insight into the knowledge discovery techniques in IoT. Finally, open research issues related to this topic have been discussed.

IX. ACRONYMS

ADAA	Advance Apriori Algorithm
ASP-tree	Associate Sensor Pattern Tree
BPs	Behavioral Patterns
CARM	Closed Association Rule Mining
CAP	Correlation Attribute Patterns
CPS-tree	Compact Pattern Stream Tree
DSARM	Data Stream Association Rule Mining
DSTree	Data Stream Tree
DSP	Differential Sensor Patterns
EDA	Event Driven Architecture
FP	Frequent Pattern
IoT	Internet of Things
IIoT	Industrial Internet of Things
G-IoT	Green Internet of Things
KDD	Knowledge Discovery in Database
MMFP	Mine Malicious Frequent Pattern
NIC	National Intelligence Council
PASP	Periodic Associate Sensor Pattern

PDS	Privacy Detection Scheme
PLT	Positional Lexicographic Tree
QoS	Quality of Service
RFSP	Regular Frequent Sensor Pattern
ShrFSP	Share-Frequent Sensor Pattern
SOA	Service Oriented Architecture
TAR	Target Association Rule
WARM	Window Association Rule Mining

REFERENCES

[1] F. Dai and J. Wu, "An extended localized algorithm for connected dominating set formation in ad hoc wireless networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 15, no. 10, pp. 908–920, Oct. 2004.

[2] A. K. M. Azad and J. Kamruzzaman, "Energy-balanced transmission policies for wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 10, no. 7, pp. 927–940, Jul. 2011.

[3] S. R. Khuntia, J. L. Rueda, and M. A. M. M. van der Meijden, "Smart asset management for electric utilities: Big data and future," in *Asset Intelligence Through Integration and Interoperability and Contemporary Vibration Engineering Technologies*. Springer, 2019, pp. 311–322.

[4] J. R. Mayaud, M. Tran, R. H. M. Pereira, and R. Nuttall, "Future access to essential services in a growing smart city: The case of Surrey, British Columbia," *Comput., Environ. Urban Syst.*, vol. 73, pp. 1–15, Jan. 2019.

[5] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[6] S. P. Mohanty, U. Choppali, and E. Kougianos, "Everything you wanted to know about smart cities: The Internet of Things is the backbone," *IEEE Consum. Electron. Mag.*, vol. 5, no. 3, pp. 60–70, Jul. 2016.

[7] R. Roshan, A. Sharma, and O. Rishi, "IoT platform for smart city: A global survey," in *Emerging Trends in Expert Applications and Security*. Springer, 2019, pp. 197–202.

[8] I. Yaqoob, E. Ahmed, I. A. T. Hashem, A. I. A. Ahmed, A. Gani, M. Imran, and M. Guizani, "Internet of Things architecture: Recent advances, taxonomy, requirements, and open challenges," *IEEE Wireless Commun.*, vol. 24, no. 3, pp. 10–16, Jun. 2017.

[9] L. Mainetti, L. Patrono, and A. Vilei, "Evolution of wireless sensor networks towards the Internet of Things: A survey," in *Proc. 19th Int. Conf. Softw., Telecommun. Comput. Netw.*, Sep. 2011, pp. 1–6.

[10] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[11] A. Iera, C. Floerkemeier, J. Mitsugi, and G. Morabito, "The Internet of Things [guest editorial]," *IEEE Wireless Commun.*, vol. 17, no. 6, pp. 8–9, Dec. 2010.

[12] Z. Ding, Q. Yang, and H. Wu, "Massive heterogeneous sensor data management in the Internet of Things," in *Proc. Int. Conf. Internet Things 4th Int. Conf. Cyber, Phys. Social Comput.*, Oct. 2011, pp. 100–108.

[13] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," *SIGMOD Rec.*, vol. 22, no. 2, pp. 207–216, Jun. 1993.

[14] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," *SIGMOD Rec.*, vol. 29, no. 2, pp. 1–12, Jun. 2000.

[15] M. F. Duarte and Y. Hen Hu, "Vehicle classification in distributed sensor networks," *J. Parallel Distrib. Comput.*, vol. 64, no. 7, pp. 826–838, Jul. 2004.

[16] S. Goel and T. Imielinski, "Prediction-based monitoring in sensor networks: Taking lessons from MPEG," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 31, no. 5, pp. 82–98, 2001.

[17] P. Desnoyers, D. Ganesan, H. Li, M. Li, and P. J. Shenoy, "PRESTO: A predictive storage architecture for sensor networks," in *Proc. HotOS*, 2005, pp. 12–15.

[18] C. Ordonez, N. Ezquerra, and C. A. Santana, "Constraining and summarizing association rules in medical data," *Knowl. Inf. Syst.*, vol. 9, no. 3, pp. 1–2, Mar. 2006.

[19] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, "Advances in knowledge discovery and data mining," *IEEE Expert*, vol. 11, no. 5, pp. 20–25, 1996.

[20] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI Mag.*, vol. 17, no. 3, p. 37, 1999.

[21] A. Milenković, C. Otto, and E. Jovanov, "Wireless sensor networks for personal health monitoring: Issues and an implementation," *Comput. Commun.*, vol. 29, nos. 13–14, pp. 2521–2533, Aug. 2006.

[22] M. Xie, S. Han, B. Tian, and S. Parvin, "Anomaly detection in wireless sensor networks: A survey," *J. Netw. Comput. Appl.*, vol. 34, no. 4, pp. 1302–1325, 2011.

[23] Y. Zhang, N. Meratnia, and P. Havinga, "Outlier detection techniques for wireless sensor networks: A survey," *IEEE Commun. Surveys Tutr.*, vol. 12, no. 2, pp. 159–170, 2nd Quart., 2010.

[24] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, p. 15, 2009.

[25] O. Boyinbode, H. Le, and M. Takizawa, "A survey on clustering algorithms for wireless sensor networks," *Int. J. Space-Based Situated Comput.*, vol. 1, nos. 2–3, pp. 130–136, 2011.

[26] A. A. Abbasi and M. Younis, "A survey on clustering algorithms for wireless sensor networks," *Comput. Commun.*, vol. 30, no. 14, pp. 2826–2841, 2007.

[27] M. M. Gaber, A. Zaslavsky, and S. Krishnaswamy, "A survey of classification methods in data streams," in *Data Streams*. Springer, 2007, pp. 39–59.

[28] A. Mahmood, K. Shi, S. Khatoon, and M. Xiao, "Data mining techniques for wireless sensor networks: A survey," *Int. J. Distrib. Sensor Netw.*, vol. 9, no. 7, Jul. 2013, Art. no. 406316.

[29] C.-W. Tsai, P.-W. Tsai, M.-C. Chiang, and C.-S. Yang, "Data analytics for Internet of Things: A review," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 8, no. 5, p. e1261, 2018.

[30] U. Ahsan and A. Bais, "A review on big data analysis and Internet of Things," in *Proc. IEEE 13th Int. Conf. Mobile Ad Hoc Sensor Syst. (MASS)*, Oct. 2016, pp. 325–330.

[31] F. Chen, P. Deng, J. Wan, D. Zhang, A. V. Vasilakos, and X. Rong, "Data mining for the Internet of Things: Literature review and challenges," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 8, Aug. 2015, Art. no. 431047.

[32] M. Marjani, F. Nasaruddin, A. Gani, A. Karim, I. Abaker T. Hashem, A. Siddiqa, I. Yaqoob, "Big IoT Data analytics: Architecture, opportunities, and open research challenges," *IEEE Access*, vol. 5, pp. 5247–5261, 2017.

[33] S. Shadroo and A. M. Rahmani, "Systematic survey of big data and data mining in Internet of Things," *Comput. Netw.*, vol. 139, pp. 19–47, Jul. 2018.

[34] C.-W. Tsai, C.-F. Lai, M.-C. Chiang, and L. T. Yang, "Data mining for Internet of Things: A survey," *IEEE Commun. Surveys Tutr.*, vol. 16, no. 1, pp. 77–97, 1st Quart., 2014.

[35] J. Tervonen, V. Isoherranen, and M. Heikkila, "A review of the cognitive capabilities and data analysis issues of the future industrial Internet-of-Things," in *Proc. 6th IEEE Int. Conf. Cognit. Infocommun.(CogInfoCom)*, Oct. 2015, pp. 127–132.

[36] P. Braun, A. Cuzzocrea, C. K. Leung, A. G. M. Pazdor, J. Souza, and S. K. Tanbeer, "Pattern mining from big IoT data with fog computing: Models, issues, and research perspectives," in *Proc. 19th IEEE/ACM Int. Symp. Cluster, Cloud Grid Comput. (CCGRID)*, May 2019, pp. 584–591.

[37] M. Habib ur Rehman, I. Yaqoob, K. Salah, M. Imran, P. Prakash Jayaraman, and C. Perera, "The role of big data analytics in industrial Internet of Things," 2019, *arXiv:1904.05556*. [Online]. Available: http://arxiv.org/abs/1904.05556

[38] P. Lade, R. Ghosh, and S. Srinivasan, "Manufacturing analytics and industrial Internet of Things," *IEEE Intell. Syst.*, vol. 32, no. 3, pp. 74–79, May 2017.

[39] L. Syed, S. Jabeen, S. Manimala, and H. A. Elsayed, "Data science algorithms and techniques for smart healthcare using IoT and big data analytics," in *Smart Techniques for a Smarter Planet*. Springer, 2019, pp. 211–241.

[40] D. V. Dimitrov, "Medical Internet of Things and big data in healthcare," *Healthcare Inf. Res*, vol. 22, no. 3, p. 156, 2016.

[41] B. Cheng, S. Zhao, J. Qian, Z. Zhai, and J. Chen, "Lightweight service mashup middleware with REST style architecture for IoT applications," *IEEE Trans. Netw. Serv. Manage.*, vol. 15, no. 3, pp. 1063–1075, Sep. 2018.

[42] J. Souza, A. Francisco, C. Piekarski, and G. Prado, "Data mining and machine learning to promote smart cities: A systematic review from 2000 to 2018," *Sustainability*, vol. 11, no. 4, p. 1077, Feb. 2019.

[43] M. M. Rashid, I. Gondal, and J. Kamruzzaman, "Mining associated patterns from wireless sensor networks," *IEEE Trans. Comput.*, vol. 64, no. 7, pp. 1998–2011, Jul. 2015.

[44] B. Cheng, S. Zhao, S. Wang, and J. Chen, "Lightweight mashup middleware for coal mine safety monitoring and control automation," *IEEE Trans. Autom. Sci. Eng.*, vol. 14, no. 2, pp. 1245–1255, Apr. 2017.
[45] J.-F. Chamberland and V. V. Veeravalli, "Wireless sensors in distributed detection applications," *IEEE Signal Process. Mag.*, vol. 24, no. 3, pp. 16–25, May 2007.
[46] K. M. Chandy, "Event-driven applications: Costs, benefits and design approaches," in *Gartner Application Integration and Web Services Summit*, vol. 2006. 2006.
[47] G. Johansson, "Configurations in event perception," in *Perceiving Events Objects*. Hillsdale, NJ, USA: Lawrence Erlbaum Associates, 1994, pp. 29–122.
[48] K. M. Alam, J. Kamruzzaman, G. Karmakar, and M. Murshed, "Dynamic adjustment of sensing range for event coverage in wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 46, pp. 139–153, Nov. 2014.
[49] S. Watanabe, *Pattern Recognition: Human and Mechanical*. Hoboken, NJ, USA: Wiley, 1985.
[50] B. Catania, A. Maddalena, and M. Mazza, "Psycho: A prototype system for pattern management," in *Proc. 31st Int. Conf. Very Large Data Bases*, 2005, pp. 1346–1349.
[51] K. M. Alam, J. Kamruzzaman, G. Karmakar, and M. Murshed, "Dynamic adjustment of sensing range for event coverage in wireless sensor networks," *J. Netw. Comput. Appl.*, vol. 46, pp. 139–153, 2014.
[52] K.-K. Loo, I. Tong, and B. Kao, "Online algorithms for mining inter-stream associations from large sensor networks," in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining*. Springer, 2005, pp. 143–149.
[53] K. Römer, "Distributed mining of spatio-temporal event patterns in sensor networks," *Proc. EAWMS/DCOSS*, 2006, pp. 103–116.
[54] S. K. Chong, S. Krishnaswamy, S. W. Loke, and M. M. Gaben, "Using association rules for energy conservation in wireless sensor networks," in *Proc. ACM Symp. Appl. Comput. (SAC)*, 2008, pp. 971–975.
[55] M. H. Le Gruenwald, "Estimating missing values in related sensor data streams," in *Proc. 11th Int. Conf. Manage. Data (COMAD)*, 2005, pp. 1–12.
[56] A. Boukerche and S. Samarah, "A novel algorithm for mining association rules in wireless ad hoc sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 19, no. 7, pp. 865–877, Jul. 2008.
[57] A. Boukerche and S. Samarah, "A new representation structure for mining association rules from wireless sensor networks," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2007, pp. 2855–2860.
[58] S. Tanbeer, C. Ahmed, and B.-S. Jeong, "An efficient single-pass algorithm for mining association rules from wireless sensor networks," *IETE Tech. Rev.*, vol. 26, no. 4, pp. 280–289, 2009.
[59] S. Tanbeer, C. Ahmed, and B.-S. Jeong, "Parallel and distributed algorithms for frequent pattern mining in large databases," *IETE Tech. Rev.*, vol. 26, no. 1, pp. 55–66, 2009.
[60] A. Appice, M. Ceci, A. Turi, and D. Malerba, "A parallel, distributed algorithm for relational frequent pattern discovery from very large data sets," *Intell. Data Anal.*, vol. 15, no. 1, pp. 69–88, Jan. 2011.
[61] M. M. Rashid, I. Gondal, and J. Kamruzzaman, "Share-frequent sensor patterns mining from wireless sensor network data," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 12, pp. 3471–3484, Dec. 2015.
[62] M. Rashid, I. Gondal, and J. Kamruzzaman, "A mapreduce based technique for mining behavioral patterns from sensor data," in *Neural Information Processing*. Springer, 2015, pp. 145–153.
[63] M. M. Rashid, I. Gondal, and J. Kamruzzaman, "Mining associated sensor patterns for data stream of wireless sensor networks," in *Proc. 8th ACM Workshop Perform. Monitor. Meas. Heterogeneous Wireless Wired Netw. (PMHWN)*, 2013, pp. 91–98.
[64] M. M. Rashid, I. Gondal, and J. Kamruzzaman, "A technique for parallel share-frequent sensor pattern mining from wireless sensor networks," *Procedia Comput. Sci.*, vol. 29, pp. 124–133, Jan. 2014.
[65] G. S. Manku and R. Motwani, "Approximate frequency counts over data streams," *Proc. VLDB Endowment*, vol. 5, no. 12, p. 1699, Aug. 2012.
[66] A. Boukerche and S. Samarah, "An efficient data extraction mechanism for mining association rules from wireless sensor networks," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2007, pp. 3936–3941.
[67] A. Boukerche and S. Samarah, "In-network data reduction and coverage-based mechanisms for generating association rules in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 58, no. 8, pp. 4426–4438, Oct. 2009.
[68] A. Das, "A novel association rule mining mechanism in wireless sensor networks," in *Proc. 3rd Nat. Conf. Emerg. Trends Appl. Comput. Sci.*, Mar. 2012, pp. 274–276.
[69] G. Wu, H. Zhang, M. Qiu, Z. Ming, J. Li, and X. Qin, "A decentralized approach for mining event correlations in distributed system monitoring," *J. Parallel Distrib. Comput.*, vol. 73, no. 3, pp. 330–340, Mar. 2013.
[70] N. Jiang and L. Gruenwald, "Estimating missing data in data streams," in *Proc. Int. Conf. Database Syst. Adv. Appl.* Springer, 2007, pp. 981–987.
[71] N. Jiang, "Discovering association rules in data streams based on closed pattern mining," in *Proc. SIGMOD Workshop Innov. Database Res.*, 2007, pp. 83–84.
[72] N. Jiang and L. Gruenwald, "CFI-Stream: Mining closed frequent itemsets in data streams," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2006, pp. 592–597.
[73] L. Pan, H. Gao, H. Gao, and Y. Liu, "A spatial correlation based adaptive missing data estimation algorithm in wireless sensor networks," *Int. J. Wireless Inf. Netw.*, vol. 21, no. 4, pp. 280–289, Dec. 2014.
[74] J. Paik, J. Nam, U. Kim, and D. Won, "Association rule extraction from XML stream data for wireless sensor networks," *Sensors*, vol. 14, no. 7, pp. 12937–12957, Jul. 2014.
[75] H. J. Woo, S. J. Shin, K. H. Joo, and W. S. Lee, "Finding context association rules over sensor-actuator data streams," *Adv. Sci. Technol. Lett.*, vol. 62, no. 7, pp. 74–77, 2014.
[76] S. Samarah, A. Boukerche, and A. S. Habyalimana, "Target association rules: A new behavioral patterns for point of coverage wireless sensor networks," *IEEE Trans. Comput.*, vol. 60, no. 6, pp. 879–889, Jun. 2011.
[77] A. Pal and M. Kumar, "Pattern generation from event oriented sensor data using distributed sensor transaction model," in *Proc. 4th Multidisciplinary Int. Social Netw. Conf.*, 2017, p. 36.
[78] H. Banaee and A. Loutfi, "Data-driven rule mining and representation of temporal patterns in physiological sensor data," *IEEE J. Biomed. Health Inform.*, vol. 19, no. 5, pp. 1557–1566, Sep. 2015.
[79] R. Rawassizadeh, E. Momeni, C. Dobbins, J. Gharibshah, and M. Pazzani, "Scalable daily human behavioral pattern mining from multivariate temporal data," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 11, pp. 3098–3112, Nov. 2016.
[80] E. Nazerfard, "Temporal features and relations discovery of activities from sensor data," *J. Ambient Intell. Hum. Comput.*, to be published.
[81] R. M. Rani and M. Pushpalatha, "Generation of Frequent sensor epochs using efficient Parallel Distributed mining algorithm in large IOT," *Comput. Commun.*, vol. 148, pp. 107–114, Dec. 2019.
[82] V. S. Kireev, A. I. Guseva, P. V. Bochkaryov, I. A. Kuznetsov, and S. A. Filippov, "Association rules mining for predictive analytics in IoT cloud system," in *Proc. Biologically Inspired Cogn. Archit. Meeting*. Springer, 2018, pp. 107–112.
[83] Z. Wang, W. Liang, Y. Zhang, J. Wang, J. Tao, C. Chen, H. Yan, and T. Men, "Data mining in IoT era: A method based on improved frequent items mining algorithm," in *Proc. 5th Int. Conf. Big Data Inf. Anal. (BigDIA)*, Jul. 2019, pp. 120–125.
[84] M.-C. Lee, J.-C. Lin, and O. Owe, "PDS: Deduce elder privacy from smart homes," *Internet Things*, vol. 7, Sep. 2019, Art. no. 100072.
[85] K. Bok, J. Jeong, D. Choi, and J. Yoo, "Detecting incremental frequent subgraph patterns in IoT environments," *Sensors*, vol. 18, no. 11, p. 4020, Nov. 2018.
[86] W. N. Ismail, M. M. Hassan, and H. A. Alsalamah, "Context-enriched regular human behavioral pattern detection from body sensors data," *IEEE Access*, vol. 7, pp. 33834–33850, 2019.
[87] W. N. Ismail, M. M. Hassan, and H. A. Alsalamah, "Mining of productive periodic-frequent patterns for IoT data analytics," *Future Gener. Comput. Syst.*, vol. 88, pp. 512–523, Nov. 2018.
[88] M. Z. A. Bhuiyan and J. Wu, "Event detection through differential pattern mining in Internet of Things," in *Proc. IEEE 13th Int. Conf. Mobile Ad Hoc Sensor Syst. (MASS)*, Oct. 2016, pp. 109–117.
[89] M. M. Rashid, J. Kamruzzaman, I. Gondal, and R. Hassan, "Periodic associated sensor patterns mining from wireless sensor networks," in *Proc. Int. Conf. Neural Inf. Process.* Springer, 2017, pp. 247–255.
[90] V. Radhakrishna, P. V. Kumar, V. Janaki, and S. Aljawarneh, "A computationally efficient approach for temporal pattern mining in IoT," in *Proc. Int. Conf. Eng. MIS (ICEMIS)*, Sep. 2016, pp. 1–4.
[91] K. Harada, Y. Sasaki, and M. Onizuka, "MISCELA: Discovering correlated attribute patterns in time series sensor data," in *Proc. 20th IEEE Int. Conf. Mobile Data Manage. (MDM)*, Jun. 2019, pp. 72–80.
[92] Z. Tianrui, W. Mingqi, and L. Bin, "An efficient parallel mining algorithm representative pattern set of large-scale itemsets in IoT," *IEEE Access*, vol. 6, pp. 79162–79173, 2018.

[93] M. M. Rashid, I. Gondal, and J. Kamruzzaman, "Regularly frequent patterns mining from sensor data stream," in *Proc. Int. Conf. Neural Inf. Process.* Springer, 2013, pp. 417–424.
[94] M. M. Rashid, I. Gondal, and J. Kamruzzaman, "Dependable large scale behavioral patterns mining from sensor data using Hadoop platform," *Inf. Sci.*, vol. 379, pp. 128–145, Feb. 2017.
[95] A. Yassine, S. Singh, and A. Alamri, "Mining human activity patterns from smart home big data for health care applications," *IEEE Access*, vol. 5, pp. 13131–13141, 2017.
[96] N. Usman, Q. Javaid, A. Akhunzada, K.-K.-R. Choo, S. Usman, A. Sher, M. Ilahi, and M. Alam, "A novel Internet of Things-centric framework to mine malicious frequent patterns," *IEEE Access*, vol. 7, pp. 133914–133923, 2019.
[97] C. Leung and Q. Khan, "DSTree: A tree structure for the mining of frequent sets from data streams," in *Proc. 6th Int. Conf. Data Mining (ICDM)*, Dec. 2006, pp. 928–932.
[98] M. M. Rashid, I. Gondal, and J. Kamruzzaman, "A novel algorithm for mining behavioral patterns from wireless sensor networks," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2014, pp. 3713–3719.
[99] S. Kniesburges and C. Scheideler, "Hashed patricia trie: Efficient longest prefix matching in peer-to-peer systems," in *Proc. Int. Workshop Algorithms Comput.* Springer, 2011, pp. 170–181.
[100] C. K.-S. Leung, Q. I. Khan, Z. Li, and T. Hoque, "CanTree: A canonical-order tree for incremental frequent-pattern mining," *Knowl. Inf. Syst.*, vol. 11, no. 3, pp. 287–311, Apr. 2007.
[101] B. Goethals and M. J. Zaki, "FIMI'03: Workshop on frequent itemset mining implementations," in *Proc. 3rd IEEE Int. Conf. Data Mining Workshop Frequent Itemset Mining Implement.*, Nov. 2003, pp. 1–13.
[102] S. Madden, "Intel Berkeley research lab data, dataset," Intel Berkeley Res. Lab, 2003.
[103] S. K. Tanbeer, C. F. Ahmed, B.-S. Jeong, and Y.-K. Lee, "Sliding window-based frequent pattern mining over data streams," *Inf. Sci.*, vol. 179, no. 22, pp. 3843–3865, Nov. 2009.
[104] M. M. Gaber, S. Krishnaswamy, and A. B. Zaslavsky, "Resource-aware mining of data streams," *J. Universal Comput. Sci.*, vol. 11, no. 8, pp. 1440–1453, 2005.
[105] M. H. S. Bangalore, "Resource adaptive technique for frequent itemset mining in transactional data streams," *Int. J. Comput. Sci. Netw. Secur.*, vol. 12, no. 10, p. 87, 2012.
[106] M. K. M. M. Rashid, M. Hossain, and B. Jeong, "ShrIO-tree: A share-frequent pattern mining approach without candidate generation," in *Proc. ICACT*, 2011, pp. 121–126.
[107] M. Z. Alam Bhuiyan, J. Wu, G. M. Weiss, T. Hayajneh, T. Wang, and G. Wang, "Event detection through differential pattern mining in cyber-physical systems," *IEEE Trans. Big Data*, to be published.
[108] P.-T. Yang, "Mining associated ranking patterns from wireless sensor networks," in *Proc. Int. Conf. Eng. Technol. Big Data Anal.*, 2016, pp. 1–5.
[109] B. Cheng, M. Wang, S. Zhao, Z. Zhai, D. Zhu, and J. Chen, "Situation-aware dynamic service coordination in an IoT environment," *IEEE/ACM Trans. Netw.*, vol. 25, no. 4, pp. 2082–2095, Aug. 2017.
[110] J. Pei, J. Han, B. Mortazavi-Asl, J. Wang, H. Pinto, Q. Chen, U. Dayal, and M.-C. Hsu, "Mining sequential patterns by pattern-growth: The PrefixSpan approach," *IEEE Trans. Knowl. Data Eng.*, vol. 16, no. 11, pp. 1424–1440, Nov. 2004.
[111] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Comput. Surv.*, vol. 46, no. 4, pp. 1–37, Mar. 2014.
[112] I. Žliobaitė, M. Pechenizkiy, and J. Gama, "An overview of concept drift applications," in *Big Data Analysis: New Algorithms for a New Society*. Springer, 2016, pp. 91–114.
[113] S. Liu, L. Feng, J. Wu, G. Hou, and G. Han, "Concept drift detection for data stream learning based on angle optimized global embedding and principal component analysis in sensor networks," *Comput. Electr. Eng.*, vol. 58, pp. 327–336, Feb. 2017.
[114] M. Basseville and I. V. Nikiforov, *Detection Abrupt Changes: Theory Application*, vol. 104. Englewood Cliffs, NJ, USA: Prentice-Hall 1993.
[115] I. Koychev, "Gradual forgetting for adaptation to concept drift," in *Proc. ECAI*, 2000, pp. 101–107.
[116] B. Krawczyk and M. Woźniak, "One-class classifiers with incremental learning and forgetting for data streams with concept drift," *Soft Comput*, vol. 19, no. 12, pp. 3387–3400, Dec. 2015.
[117] B. Hammer, "Incremental learning algorithms and applications," in *Proc. ESANN*, 2016, pp. 1–13.
[118] A. Lorentz. (2013). *With Big Data, Context is a Big Issue*. [Online]. Available: http://www.wired.com/insights/2013/04/with-big-data-context-is-a-big-issue/
[119] O. Vermesan, P. Friess, P. Guillemin, S. Gusmeroli, H. Sundmaeker, A. Bassi, I. S. Jubert, M. Mazura, M. Harrison, and M. Eisenhauer, "Internet of Things strategic research roadmap," *Internet Things-Global Technol. Societal Trends*, vol. 1, pp. 9–52, Jan. 2011.
[120] N. Li, V. Varadharajan, and S. Nepal, "Context-aware trust management system for IoT applications with multiple domains," in *Proc. IEEE 39th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jul. 2019, pp. 1138–1148.
[121] L. Chen, L. Duan, A. Anpalagan, O. A. Dobre, and Z. Niu, "Sustainable green networking and computing in 5G systems," *IEEE Wireless Commun.*, vol. 24, no. 4, pp. 12–13, Aug. 2017.
[122] C. Zhu, V. C. M. Leung, K. Wang, L. T. Yang, and Y. Zhang, "Multi-method data delivery for green sensor-cloud," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 176–182, May 2017.
[123] F. K. Shaikh, S. Zeadally, and E. Exposito, "Enabling technologies for green Internet of Things," *IEEE Syst. J.*, vol. 11, no. 2, pp. 983–994, Jun. 2017.
[124] R. Ahmad, M. A. Asim, S. Z. Khan, and B. Singh, "Green IoT—Issues and challenges," in *Proc. Int. Conf. Adv. Comput. Softw. Eng. (ICACSE)*, 2019, pp. 1–5.
[125] D. A. Bashar, "Review on sustainable green Internet of Things and its application," *J. Sustain. Wireless Syst.*, vol. 1, no. 4, pp. 256–264, Jan. 2020.
[126] V. Tahiliani and M. Dizalwar, "Green IoT systems: An energy efficient perspective," in *Proc. 11th Int. Conf. Contemp. Comput.*, Aug. 2018, pp. 1–6.
[127] S. K. Datta, R. P. F. Da Costa, C. Bonnet, and J. Härri, "Web of things for connected vehicles," in *Proc. Int. World Wide Web Conf.*, 2016, pp. 1–3.
[128] E. Bertino and E. Ferrari, "Big data security and privacy," in *A Comprehensive Guide Through the Italian Database Research Over the Last 25 Years*. Springer, 2018, pp. 425–439.
[129] O. Arias, K. Ly, and Y. Jin, "Security and privacy in IoT era," in *Smart Sensors at the IoT Frontier*. Springer, 2017, pp. 351–378.
[130] J. Zhou, Z. Cao, X. Dong, and A. V. Vasilakos, "Security and privacy for cloud-based IoT: Challenges," *IEEE Commun. Mag.*, vol. 55, no. 1, pp. 26–33, Jan. 2017.
[131] R. v. d. M. Janessa Rivera. (2013). *Internet of Things*. [Online]. Available: http://www.gartner.com/newsroom/id/2636073
[132] A. Costin, J. Zaddach, A. Francillon, and D. Balzarotti, "A large-scale analysis of the security of embedded firmwares," in *Proc. 23rd USENIX Secur. Symp.*, San Diego, CA, USA, Aug. 2014, pp. 95–110. [Online]. Available: https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/costin
[133] F. F. C. Center. (2013). *Internet of Things–Privacy and Security in a Connected World*. [Online]. Available: https://www.ftc.gov/news-events/events-calendar/2013/11/internet-things-privacy-security-connected-world
[134] Z. Yin, W. Liu, and S. Chawla, "Adversarial attack, defense, and applications with deep learning frameworks," in *Deep Learning Applications for Cyber Security*. Springer, 2019, pp. 1–25.
[135] A. Ur-Rehman, I. Gondal, J. Kamruzzuman, and A. Jolfaei, "Vulnerability Modelling for Hybrid IT Systems," in *Proc. IEEE Int. Conf. Ind. Technol.(ICIT)*, Feb. 2019, pp. 1186–1191.

MD. MAMUNUR RASHID (Member, IEEE) received the Ph.D. degree in computer science from Monash University. He is currently a Lecturer with the School of Engineering and Technology, CQUniversity Melbourne, Australia. His research interests include data mining and knowledge discovery from wireless sensor networks, big data analytics, network security, and distributed computing.

JOARDER KAMRUZZAMAN (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees in electrical and electronic engineering from the Bangladesh University of Engineering and Technology, Dhaka, and the Ph.D. degree in information systems engineering from the Muroran Institute of Technology, Hokkaido, Japan.

He is currently a Professor with the School of Science, Engineering and Information Technology, Federation University Australia. He has published more than 240 peer-reviewed publications which include 70 journal articles, 160 conferences, 11 book chapters, and two edited reference books. His publications are cited more than 2360 times and have H-Index 22, G-Index 40, and I-10 Index 61. He has received nearly A$2.3m competitive research funding, including prestigious Australian Research Council (ARC) Grant and large Collaborative Research Centre (CRC) Grant, and successfully supervised 20 Ph.D.s and eight masters to completion. His research interests include distributed computing, the Internet of Things, machine learning, and cyber security. He was a recipient of Best Paper Award in four international conferences ICICS'15, Singapore; APCC'14, Thailand; IEEE WCNC'10, Sydney, Australia; and in the IEEE-ICNNSP'03, Nanjing, China. He was the founding Program Co-Chair of the first International Symposium on Dependability in Sensor, Cloud, and Big Data Systems and Applications (DependSys), China, in 2015. He has served 32 conferences in leadership capacities, including the program co-chair, the publicity chair, the track chair, and the session chairs, and since 2012 as the Editor of the *Journal of Network and Computer Applications* (Elsevier). He had served as the Lead Guest of the *Journal Future Generation Computer Systems* (Elsevier) and a Guest Editor of the *Journal of Networks*.

MOHAMMAD MEHEDI HASSAN (Senior Member, IEEE) received the Ph.D. degree in computer engineering from Kyung Hee University, South Korea, in February 2011. He is currently an Associate Professor with the Information Systems Department, College of Computer and Information Sciences (CCIS), King Saud University (KSU), Riyadh, Saudi Arabia. He has authored and coauthored around more than 180 publications, including refereed IEEE/ACM/Springer/Elsevier journals, conference papers, books, and book chapters. Recently, his four publications have been recognized as the ESI Highly Cited articles. His research interests include cloud computing, edge computing, the Internet of Things, body sensor networks, big data, deep learning, mobile cloud, smart computing, wireless sensor networks, 5G networks, and social networks. He was a recipient of a number of awards, including Best Journal Paper Award from IEEE Systems Journal in 2018, Best Paper Award from CloudComp in 2014 Conference, and the Excellence in Research Award from King Saud University (two times in row, 2015 and 2016). He has served as the chair and a Technical Program Committee member in numerous reputed international conferences/workshops, such as IEEE CCNC, ACM BodyNets, IEEE HPCC, and so on.

SAKIB SHAHRIAR SHAFIN is currently pursuing the Bachelor of Science degree in electrical and electronic engineering with the Islamic University of Technology, Gazipur, Bangladesh. He is also with the Network and Data Analysis Research Group, Islamic University of Technology. His research interests include the Internet of Things (IoT), sensor networks, and real-time control, and telemetry systems.

MD. ZAKIRUL ALAM BHUIYAN (Member, IEEE) received the B.Sc. degree from International Islamic University, Chittagong, Bangladesh, in 2005, and the M.Eng. and Ph.D. degrees from Central South University, China, in 2009 and 2013, respectively, all in computer science and technology. He is also a Member of the Center for Networked Computing (CNC). Earlier, he worked as a Postdoctoral Fellow at the Central South University, China, a Research Assistant at the Hong Kong PolyU, and a Software Engineer in industries. He is currently an Assistant Professor (Research) with the Department of Computer and Information Sciences, Fordham University. His research focuses on dependable cyber physical systems, wireless sensor network applications, network security, and sensor-cloud computing. He is a member of ACM. He has served as a managing guest editor, the program chair, the workshop chair, the publicity chair, a TPC member, and a reviewer of international journals/conferences.