Privacy Preserving Data Mining Approach for IoT based WSN in Smart City

Ahmed M. Khedr[1]
Computer Science Department,
University of Sharjah, Sharjah 27272, UAE

Walid Osamy[2]
Faculty of Computers and Artificial Intelligence,
Benha University, Benha, Egypt.
Qassim University, Buridah, KSA

Ahmed Salim[3]
Math. Department, Zagazig University,
Zagazig, Egypt.
Qassim University, Buridah, KSA

Abdel-Aziz Salem[4]
Department of Basic Science, Faculty of Computers
and Informatics, Suez Canal University.
Qassim University, Buridah, KSA

***Abstract*—Wireless Sensor Network (WSN) is one of the most fundamental technologies of Internet of Things (IoT). Various IoT devices are connected to the internet by making use of WSN composed of different sensor nodes and actuators, where these sensor nodes collaborate and accomplish their tasks dynamically. The main objective of deploying WSN-based applications is to make high precision real-time observations, and it is extremely challenging because of the limited computing power of the sensors operating under constrained environments, resource constraints like energy, computation speed, bandwidth and memory, huge volume of high speed, heterogeneous and fast-changing WSN data. These challenges encouraged the researchers to concentrate deeper on exploring data mining techniques to extract the required information from the fast-changing sensor data in WSN and thereby efficiently handle the massive data generated by the WSNs. The increasing need of data mining techniques for WSN has inspired us to propose a distributed data mining technique that effectively handles the data generated by the nodes in the WSN and prolongs the lifespan of the network. Our work provides a novel cluster based scheme to mine the sensors data without moving it to cluster head (CH) or base station (BS) to achieve maximum performance in a WSN environment. The basic idea of the proposed work is that local computations are performed by utilizing the computing power at each sensor node and then the minimum higher level statistical summaries are exchanged, which decreases the energy dissipation in communication as the amount of the sensor data transferred is considerably reduced, and thereby the sensor network lifetime is maximized and also preserve the privacy of the sensor data.**

***Keywords*—Distributed cluster-based algorithm; association rules; Internet of Things (IoT); privacy preserving; vertically and horizontally distributed databases; wireless sensor networks (WSN)**

I. Introduction

Knowledge discovery from sensor data is an emerging research area and mining data efficiently and effectively from the resource constraint sensor nodes is really challenging. Recently, the researchers on data mining are strongly motivated by the challenges raised by the huge volume, high speed, heterogeneous, rapid and fast-flowing data generated by WSNs and gathering crucial information from the device data has become a topic of active research. As a result, new data mining techniques has been developed and some of the existing approaches has been modified to devise new analytic methods for the massive quantity of data generated from sensor networks. Distinct data mining techniques that deal with extraction and analysis of WSNs data concentrated on clustering [1], association rules [2], [3], frequent patterns [4], [5], sequential patterns [6] and classification [7] have been efficiently applied on sensor data to make intelligent WSN applications. Most of the traditional mining techniques perform intensive centralized computation and is expensive. Moreover, the deployment and implementation of WSNs create a lot of research challenges making the direct application of traditional mining techniques inappropriate to WSN. The huge volume and high cost of storage make it quite impossible to store the fast-flowing WSN data or to inspect it several times. The nature and characteristics of the sensor data, limited communication and computation capabilities of the sensor, and special design and deployment limitations of the WSNs make the application of primitive data mining procedures challenging. Therefore, it is highly required to devise data mining techniques that are capable of handling continuous, fast-changing and extensive data streams of WSN with high dimensionality and distributed nature, and to analyze and process it in single-pass, multi-level, multi-dimensional or online methods of data mining.

With the rapid advancement of sensor network technology, WSN based systems are becoming more popular and are increasingly finding its applications in different areas of knowledge [8]. This resulted in the generation of diverse WSN applications yielding extensive heterogeneous distributed data to be analyzed efficiently and effectively [9]. Such applications are mostly critical and require real-time control and reliable operation as crucial demands.

WSN acts as a virtual layer and has become an intrinsic part of IoT in a secure manner. But to do so, it has to overcome various challenges such as security, integration issues, energy optimization, network lifetime, etc. WSN is like the eyes and ears of the IoT. It is the link which joins the real world to the digital world. And it is also responsible for passing on the sensed real world values to the Internet. IoT based wireless sensor networks (IoT-WSNs) have a wide range of applications in various fields, which allows inter connection

of different objects and nodes through Internet. IoT-WSN can be described as collection of enormous sensor nodes deployed with a number of moving objects or devices (such as intelligent cars) over a large area (Smart City) to sense and accumulate various data from the environment and systems for different applications such as weather monitoring, animal tracking, disaster management, bio-medical applications.

It has been proved that a database oriented approach is helpful to manage the dynamic nature of WSN data for those applications [10], [11] and hence it motivated the researchers to treat WSN as a distributed database. Accordingly, WSN can be modelled as a distributed database where sensor devices act as data sources and stores the data with the sensor in the form of database across the network in distributed manner [12], [13]. The main objective of distributed database management [14] on WSNs is to facilitate the energy efficient analysis of massive data sensed by the sensor nodes. In order to extract the data, a WSN database should provide support of robust queries eg.: SQL-like abstractions for interaction with the network [15].

Upon research on sensor hardware, it has been proved that the depletion of energy in WSN [16] is mainly during the exchange of data among the sensor nodes. Different data reduction techniques [17], [18] such as packet merging, aggregation [19], approximation based techniques [21]], data compression techniques [20], [23], [24], [25], [26], [27], [28] and data fusion [22] techniques are being used to handle this problem.

Privacy preserving data aggregation is a challenge in IoT-WSN as it could be spied. The privacy preserving data aggregation has intent to save individual privacy of the nodes using transmission directions, in such a way that the enemies cannot obtain the sensitive information of a specific node. A city is considered smart city when it functions in a sustainable and intelligent way through the integration of all its infrastructures and services by deploying intelligent devices and networks for monitoring and control. Smart city applications such as intelligent transportation systems as well as monitoring public infrastructures (e.g. bridges, roads and buildings) are based on the data aggregation by static and mobile sensors deployed in very large numbers. We handle the data' privacy node at aggregation time by considering that privacy preservation can be achieved if nodes have devices that may be heterogeneous and have different places as a result every node only sense a small part of the whole collected data; their results have to be aggregate to give a whole and global scenario (IoT-WSN will be considered as big distributed database system with vertical partitioning data). This provides fast query processing and preserving the individual sites private data.

We propose a new version of association rule algorithm to work with distributed IoT based WSNs data. This paper provides a methodology in which each CH is represented by an agent, with the potential to decompose global computations into local ones. That means, the computation at CH is executed by exchanging minimal statistical summaries with other agents at sensor nodes and the results of every CH will be globally aggregated at the BS. The main objective of using this distributed approach is to lessen the data transfer and energy utilization of sensors upon exchanging data with central server, which conserves energy and prolongs the lifespan of network to a great extent.

The rest of this paper is divided and structured as follows: The related research of the proposed problem is described in Section II, while integration methodology of the databases is given in Section III. Section IV presents the proposed scheme. The simulation results and comparison with other baseline algorithms are included in Section V and finally, conclusion of the paper is given in Section VI.

II. Related Research

WSNs producing huge volume of data is offering a promising prospect for data analytic and mining techniques to involve in extracting useful information for a wide variety of applications. Existing data mining techniques adopted for WSN are classified depending on its application: whether it is on network side or central side. Sequential mining, frequent mining, clustering and classification are the four classes of data mining approaches commonly adopted in WSNs which use both centralized and distributed approaches. The mining techniques based on these above mentioned classes form the first type of classification. Most of the techniques based on sequential and frequent pattern mining are adaptation of the primitive techniques such as the FP-Growth Algorithm and the influential algorithm like Apriori for association rule determination and learning for extracting potentially valuable information from large amounts of WSN data, while the techniques based on clustering have adapted the data correlation-based clustering, k-means and the hierarchical clustering approaches. Approaches based on classification have adapted several major kinds of classification techniques including k-nearest neighbor classifier, Bayesian networks, Decision tree, Neural Networks and Support Vector Machines, according to the type of classification model adopted. The second type of classification is based on how the data is processed and analyzed- either centrally or in distributed way. The limited computing power of the sensors operating under constrained environments, resource constraints like energy, computation speed, bandwidth and memory, huge volume of high speed, heterogeneous and fast-changing data generated by WSNs, make distributed processing a more preferable solution. The distributed data computation approaches perform mining and other processing of data at sensors locally and then accumulate the results. The third type classification is based on the attitude concerned with solving a specific problem. This level of classification is mainly focused on WSNs performance issues and application issues. The characteristics of the sensor data, precision, accuracy and real-time decision making of the WSN applications often require abundant use of communication and energy. The algorithm proposed in this paper is a distributed frequent pattern mining technique with reduced energy consumption and improved WSN lifetime.

According to the above classification of mining algorithms, we review the existing literature as follows. Frequent pattern mining technique finds the stream of data that frequently occurs in the dataset, with the objective to determine crucial relations existing between the sensors data. Primitive algorithms for frequent pattern mining [29], [30], [31], [32], [33], [34], [39] are resource intensive and cannot be directly used for mining the fast-flowing WSN data. Numerous algorithms are designed for solving the application-based issues of WSNs such as [35], [36], [37], [49]. In this context, the authors of [35] introduced a centralized technique named Data Stream

Association Rule Mining (DSARM) to determine the missed values in sensor data readings due to corruption or loss of messages. It identifies sensors that repeatedly report the same data and estimates the missed values in sensor data by making use of the data communicated by its related sensors. Closed item-sets-based Association Rule Mining (CARM) is a data estimation technique [36], [37] for deriving the recent sensor association rules based on the latest closed item-sets in the sliding window. Other than these, a number of centralized algorithms were also designed to maximize the performance of WSNs [38], [40], [41]. The authors of [38] proposed online one-pass methodology, in which the WSN data-stream is converted to interval list (IL), for inter-stream association rule mining from bulk sensor data stream. A rule-learning model is proposed in [40] to derive powerful rules from sensor data readings, to control and coordinate WSN operations. The sensor pattern tree proposed in [41] is a tree-based data structure for deriving association rules from sensor data by scanning the database only once.

Several distributed approaches exist in the literature for solving application based issues of WSNs and/or maximizing the performance of WSN [43], [2], [42]. Author in [43] presents a distributed method with some spatial/temporal properties to detect frequent event patterns. Their work describes an in-network approach of data mining in which user can specify the spatial and/or temporal proximity for patterns among events in which he is interested in. The sensors collect events accordingly and execute a data mining procedure to identify the pattern among these collected events that satisfies the user specified parameters. The frequent patterns obtained after mining are then converted into association rules. Every node in the network will then send the discovered local patterns to the sink for performing secondary mining on these patterns to create a global picture of the entire network with respect to time and space. However, the communication overhead of event collection and memory consumption of item-set discovery algorithm are the major issues with this methodology. A distributed data extraction methodology is presented in [2] to accumulate the data on sensor in an attempt to lessen the number of message exchanges. Each node in the network is equipped with a buffer and has corresponding entry for support value. Moreover, parameters like time-slot size, support, and historic period are distributed across the WSN. The sensor node will examine the received messages per time slot and sets its buffer entry accordingly. Every node checks its buffer upon completion of historic period and compares the set value with the initially provided support value. The message will be transferred to the sink if the set value is larger or equal to the support value. The potential drawbacks of this technique include delay in critical messages for high support value and the node buffer cost. The authors of [42] proposed a new representational structure named Positional Lexicographic Tree (PLT). Using the specified sensor allocation rules for the event detecting sensors, it stores the sensor's event-detecting status and is a promising structure that can be used for indexing and compressing the sensor data residing in any transactional database. It also facilitates subset checking using data summaries. The way in which the PLT identifies the conditional structure is found to be comparatively easier than FP-tree method. PLT also allows the mining process to be partitioned into various tasks; each of which can be accomplished separately. The issue with PLT is that it requires multiple scans and PLT updates, restricting the effective use of this technique in dealing with WSN data.

Our proposed approach is completely different from the above and to the best of our knowledge this is a new approach to handle the mining problem in WSNs. The key idea of our distributed approach is to treat the whole network as a distributed database where, the sensed data is kept at the sensor nodes, stored in rows, and the columns represent sensor attributes. We do not move the sensor data to the CH or BS. We consider the global data D in implicit format, i.e., the tuples belonging to sensor database D is distributed over all the nodes in the WSN, with each node generating and accumulating its data. In our approach the sensor nodes in the network are grouped into various clusters, each having a CH managing the cluster. The queries generated by the CH will be periodically answered by the CMs belonging to it, based on their readings. Only the statistical summaries will be transmitted to CHs. It accumulates the received summaries and transfers the aggregated computation output to the BS for final results. Rather than traditional queries which mainly focus on the current state of a database, these queries (SQL-like abstractions) are often continuous so that the application will be notified continually about the changes recorded by the sensors [12]. A sensor node sends its response to the current query only if its reading is different from the last recorded reading. This method of aggregation efficiently decreases the huge volume of data transmitted through the network, reducing the computation burden enforced on the BS, which in turn raises the lifespan of the WSN. Furthermore, for efficient execution of query with low energy dissipation and delay, an effective load balancing policy can be adopted in terms of remaining power and the load of the nodes.

III. Databases Integration

As discussed before, each sensor node is assumed to have a relational database with a number of attributes. In this section, we describe different scenarios of databases and the proposed methodology to handle these databases.

A. Nature of Data Distribution

Vertically Distributed Databases: The implicit database D exists as a set of distributed fragments across all the devices in the WSN. Each database component D_i stored at device node (s_i) consists of a set of records (tuples), where every record stores a diverse attribute set. Another component D_j stored at device node s_j $j \neq i$ could contain some attributes shared in common with D_i, as well as distinctive attributes that aren't shared with other components. The databases following vertical strategy of distribution need "Join" operation on all explicit D_i's to build the implicit database D. The planned algorithm will make use of the shared attributes in performing data processing. This approach demonstrates a more practical scenario than using a single key, non-overlapping set of attributes for the components distributed across all the devices in the WSN. Our aim is to enable the participation of these independently designed local component databases to have arbitrary overlapping set of attributes upon collaboration with each other resulting in single global database. The association rule is performed on the database resulting from joining the

relations on the sensor devices distributed across the network. *Horizontally Distributed Databases:* The implicit database D exists as a set of distributed fragments $D_1, D_2, \ldots, D_n$ across the sensor nodes $s_1, s_2, \ldots, s_n$ such that every D_i contains unique set of tuples having identical attributes set. The set of tuples present in the distributed D_i's will then be merged together to form the global implicit relation D. The proposed methodology is applicable on any of the above mentioned distribution of database, either horizontal or vertical.

B. Problem Formulation

The entire WSN is treated as a distributed database such that each sensor node has a local component database following either horizontal or vertical data distribution. The join of these databases at the BS or end user creates the global implicit database D containing significant data for computation and mining tasks. The key objective of the proposed algorithm is to reduce the data exchanges by retaining the local data with the node itself. Only the local results of D_i of CMs will be transmitted to be aggregated at their CH and then the aggregated results of CHs will be transmitted to be aggregated at BS. Let A be the attribute set of the global implicit database D and A_i be the attribute set of the local database D_i at sensor node s_i. Then A can be defined as the union of all the attributes between all the local databases within the network.

$$A = \bigcup_{i=1}^{n} A_i \tag{1}$$

The subset of attributes shared in common between local databases D_i and D_j can be represented as S_{ij} such that

$$S_{ij} = \bigcap_{x=i,j} A_x \tag{2}$$

The union of all attributes shared in common with all local databases forms the shared attribute set S of D.

$$S = \bigcup_{i,j} S_{ij} \tag{3}$$

That means, S contains all the shared attributes within the network.

Our aim is to determine the association rules of global implicit database D with minimum message exchanges. So, the global computation is decomposed into local ones considering the shared attribute constraints as well as preserving the data privacy. Summaries of the local computation will then be aggregated to derive the global association rules. The constraints enforced on sharing attributes among the agents help in ensuring data privacy and confidentiality.
The mathematical formulation of the proposed problem can be described as follows: Let F be a function applied on D and the result be denoted as R, such that,

$$R = F(D). \tag{4}$$

As mentioned earlier, the desired distributed computation here is to derive association rules related to the database D, R denotes the derived associated rule and F denotes the algorithm implementation for deriving R from database D. The responsibilities of an agent is to execute local processing on its database fragment and to communicate with other agents, exchanging local processing results to accomplish the global computation. The shared attribute set S will ultimately result in determining the implicit D obtained from the explicitly involved partitions D_1 to D_n. The functionally equivalent implementation of F (in equation 4) is defined as follows:

$$R(S) = H[h_1(D_1, S), h_2(D_2, S), \ldots, h_n(D_n, S)]. \tag{5}$$

Here, $h_i(D_i, S)$ denotes the local data computation executed by i^{th} agent on database D_i residing on sensor s_i. S denotes the shared attributes and H denotes the aggregation operation upon local computation results, performed by the CH. Each problem requires unique set of h-operators as the count of h_i and the characteristics of both operations H and h_i depend on the shared attribute set S and the involved D_is. The objective of data privacy presented in our method is to prevent attacker/intruder from figuring out any record/tuple at any node in the network and this is made possible by the effective use of hash functions and aggregation methods.

IV. Distributed Mining Association Rules from WSNs

An association rule can be defined as an implication, A implies B, represented as $A \Rightarrow B$, where A, B denotes the item-sets, referred as antecedent and consequent respectively. That means, the transaction records in the database including items in item-set A should also be including items in item-set B. In the proposed technique, determining the association rules in D is the required global computation. It is decomposed and distributed across the network, therefore computation is locally performed and the needed statistical summaries are then collected and exchanged. To initiate the global computation, BS starts the process by requesting the CHs to compute tasks such as support and confidence, then agent at each CH sends request to begin the local computation to its CMs. This will decrease the number and size of messages communicated to and from CMs and their respective CHs and also between BS and CHs, which in turn decrease the consumed energy and prolong the lifespan of the network. Furthermore, this algorithm preserves the data privacy during communication. The proposed distributed algorithms for association rule is composed of three major phases: Initialization, Support and Confidence Computing, and Aggregation. In Initialization phase, every CH creates the relation Shared using shared attributes and shared values from its members. In Support and Confidence Computing phase, CHs initiate the queries and compute the Support and Confidence and send the results to BS and in Aggregation phase, the BS will find the final association rules.

After the WSN is divided into k clusters using any clustering algorithm such as DEC, each cluster head CH_i has m CMs ($s_j^i, j = 1...m$, information of its CMs such IDs, locations, attributes they measure, etc). The CH_i and its members will cooperate with each other and execute mining algorithm as follows:

A. Initialization Phase

We define a relation called P-shared on the attributes in the set S. This relation, P-shared contains tuples related to

all combinations of values possible for the attributes in S. The relation P-shared would have mediated the creation of the explicit D, if it was attempted and is used by us in a very similar role. Then, the tuples having zero count at each node will be removed from P-shared and the resulting relation is known as Shared relation.

1) Every CH_i creates *P-shared* relation. The attributes of *P-shared* are the attributes in S and the tuples are the cross product of distinct shared values.
2) Then, generate the *Shared* relation from *P-shared* as follows:
   - Each CM receives s_j^i replies to the $P - Shared$ message by the array $Count - of - Tuples$ that contains count of each tuples.
   - CH_i removes all tuples that have zero count and form *Shared* relation.
   - Index the *Shared* relation beginning with zero

B. Support and Confidence Computing Phase

Support and Confidence Computing phase will be executed by every CH which performs the following two functions:

1) Maintaining the active item-sets and enumerating the candidate sets at the succeeding level from the frequent item-sets in the preceding level.
2) Computing the support and confidence levels.

Implementation of first part can be briefed as follows: the agent on every CH initiates implementation of the algorithm; this agent will carry out the major control tasks of the algorithm, such as finding and managing the active as well as the candidate item-sets, interacting with agents on CMs to compute the two significant measures of association rule namely support and confidence. The computation task is thus decomposed and the process is repeated iteratively and controlled by the agents.

The ratio of the count of transactions containing the item-set to the total transaction count in D is termed as the support of an item-set. Thus, the essential computational primitive in the described technique is to find the total number of tuples in D and it can be calculated only after getting the local computational results from each node. However such computations fulfilling specific attribute-value conditions are a bit complicated and is described as follows.

1) Count of Tuples in Implicit Database: The tuples of global database D are implicit, and are not explicitly visible in a relation, making it more difficult to find the number of enclosed tuples. In our case, we decompose this process of finding tuples, requesting feedback from the agents of D_i's regarding the local counts. The corresponding replies from the agents are then used to determine $N_{total}(D)$, i.e., the total tuples in D. This can be formulated as given below

$$N_{total}(D) = \sum_j \prod_t (N_{D_t})_{cond_j} \tag{6}$$

Here, the subscript $cond_J$ defines the attribute-value condition of the j^{th} tuple of relation Shared, t denotes the index of cluster member and $N(D_t)_{cond_J}$ is the count of tuples in relation D_t satisfying $cond_j$. According to Equation 5, we can have:

$$h_i(D_i, S) = N(D_i)_{cond_j} \tag{7}$$

Such that j refers to the j^{th} tuple of relation Shared. It is required to have such summary for each tuple in Shared from each agent. In order to decrease the interaction between agents, relation Shared can either be managed and maintained by one agent or by each agent separately. The role of the function H is to calculate the sum-of-products from the deduced summaries according to the Equation 6, in which each product term represents the count of tuples satisfying $cond_j$ in a D_i and the resultant gives the number of distinct tuples fulfilling $cond_j$, needed for the implicit Join of all the D_i's. Then the summation operation is performed on the product terms computed for each tuple. This operation simulates a Join operation executed on all the databases without explicit enumeration of the tuples. The most favorable aspect of decomposing $N_{total}(D)$ is that it is possible to translate each product term $N(D_t)_{cond_J}$ into an SQL query; select count (*), such that $cond_j$ can be executed by the local agent at D_t.

2) Support and Confidence for Candidate Sets: The ratio of the count of transactions containing the item-set to the total transaction count in D is termed as the support of an item-set. Also, confidence with respect to a set of transactions refers to the proportion of the transactions that contains A which also contains B. Thus, the essential computational primitive in the described method is to find the total number of tuples in D and it can be calculated only after getting the local computational results from each node. It is possible to extend the decomposition for count into count of tuples satisfying a new condition by changing $cond_j$ in Equation 6 as shown below:

$$N_{new-condition} = \sum_j \prod_{t=1}^{n} N(D_t)_{cond_j \ and \ new-condtion} \tag{8}$$

The above equation is necessary to find the support level for a candidate frequent item-set. The new condition is formed by the attribute values in the frequent item-set. The method in which an agent finds the support measure for a candidate frequent item-set is as described below. In relation Shared, the agent checks the condition specified and identify the tuples matching the attribute-value pairs in the candidate set and then retains those tuples to find the number of tuples resulted from this reduced Shared relation. The support level for a candidate set of attribute-value pairs is given by the ratio of the resultant candidate set count by the total count N_{total}.

The algorithms for generating candidate item-sets and computing the frequent item-sets at every CH, which form the common computational primitives.

C. Extract and Integration of Rules Phase

Extraction: Every CH_i extracts the association rules using frequent item-sets F. The main steps of the extracting rules procedure will be as follows:

- for every frequent item-set $f \in F$, using all nonempty subsets c of f and $c \neq f$.

- for every subset c, if $\frac{support(f)}{support(c)} \geq confidence$
- $R_i = R_i \bigcup \{c \Rightarrow (f - c)\}$

Using Equation 6, calculate the total number of tuples ($no\_of\_tuples = \sum_j \prod_i Count-of-Tuples[i][j]$) and send message containing the set of rules R_i with confidence of each rule and no_of_tuples to the BS.

Integration: The BS receives the set of rules R_i including the confidence of each rule and the size of the implicit database N_i at CH_i ($i = 1, 2, \ldots, k$). The BS integrates the rules by weighing the confidence of the rules using the N_i and the total number of tuples in the whole database of the network. The main steps of the integration procedure will be as follows:

- Input R_i=$\{r_i^j, c_i^j\}$, $i = 1, 2, \ldots, k$ and j is the number of rules in R_i and c_i^j is the confidence or rule r_i^j.
- N_i is the number of tuples received from CH_i.
- Assume δ is the total number of tuples received from the k clusters.
- $R = R_1 \bigcup \cdots \bigcup R_k$.
- for each $r_l^m \in$ R
- for $i = 1$ to k
- if r_l^m in R_i with confidence c_i^t
- c_l^m= c_l^m + (N_i/δ) * c_i^t

D. Complexity Computing and Analysis

The cost of working with implicitly specified set of tuples can be measured in various ways. One cost model computes the number of messages that must be exchanged among various sites. Complexity for distributed query processing in databases has been discussed in [44] and this cost model measures the total data transferred for answering a query. In our case the amount of data transferred is very little (statistical summaries) but the number of messages exchanged may grow rapidly with the number of iterations of the proposed mining algorithm [45], [46], [48]. At each cluster in the network, in order to extract the association rules, a number of messages need to be exchanged. Let us say:

1) the number of Cluster Heads is K.
2) the average number of members in each cluster is m nodes.
3) we have k -frequent item sets.

We derive below an expression for the number of messages that need to be exchanged for our proposed algorithm dealing with the implicit set of tuples.

Creating the shared relation: the number of messages to create the shared relation can be summarized as follows:

- $m+1$ messages for requesting and receiving the different items of shared attributes.
- $m+1$ messages from CH to members contains Pshared and the reply from members.
- 1 message from CH to members containing the indexed shared relation.

The total number of message per cluster is $3+3m$, i.e., for K clusters it will be $K(3 + 2m)$.

Finding 1^{st} frequent item set: Assume in each cluster, each node has c unshared attributes then we have $c\ m$ unshared attributes. In order to get 1^{st} frequent item, there are two types of items shared and unshared. No messages are needed for shared items where it can find 1^{st} frequent sets using the shared relation at the CH. While for unshared items, CH requests the count of distinct unshared items at each member to get 1^{st} frequent. Therefore, unshared case requires $2c\ m$ messages ($c\ m$ requests to members and $c\ m$ replies from members).

Finding $k-$ frequent item set: As we discussed in algorithm and in example scenario above, each k-frequent item-sets can be in one of the following cases: fully shared, fully unshared and partially shared.

- In case of fully shared, the k-frequent item-sets will be computed at CH, i.e., no exchanged messages are required.
- In case of fully unshared, $2 * k * f_1$ messages are needed to compute the k-frequent item-sets, where f_1 is the average number of frequent item-sets that are fully unshared.
- In case of partially shared, $\sum_{k=2}^{l} 2(k-1)f_2$ exchanged messages are needed to find the k-frequent item-sets in the worst case, where f_2 is the average number of partially shared items and l is the frequent length. In worst case, we have $k-1$ unshared item-sets, each item-set requires $2(k-1)$ exchanged messages to be sent (requests to members and the replies from members).

  Therefore, the total number of messages will be:

$$Total\ \ messages = K(2m + 3 + 2c\ m + \sum_{k=2}^{l} 2(k-1)f_2 + 2kf_1). \quad (9)$$

V. Simulation Results

In this section, using MATLAB R2016b, we validate feasibility and evaluate the performance of the proposed algorithm. Two types of experiments are performed for validation and evaluated the effectiveness of the proposed approach on the base of DEC [47] as clustering algorithm for WSNs. We test the effect of support value, percentage of CHs and percentage of shared attributes on the number of exchanged messages.

A. Exchanged Messages Parameters

In this set of test 100 sensor nodes are deployed randomly on a 2D-plane to monitor a region with size $100 \times 100\ m^2$. The simulation results are gained by averaging with different topology seeds and 10 clusters (except in the performance evaluations, we vary the number of clusters).

Support value: In the first test, we show the effect of the selected support values on the number of exchanged messages. The support value is varied from 0.1 to 1 with incremental of 0.1 and at each value the number of messages are computed. Fig. 1 shows that the number of messages decreases as the support value increases. The reduction of the number of messages is from 40% to 90% compared to the centralized methodology (or centralized extraction) in which all data transferred to CH, i.e., with zero support value. Therefore, the proposed mining vertical data approach reduces the amount of transferred data and as a result, decreases the number of messages.

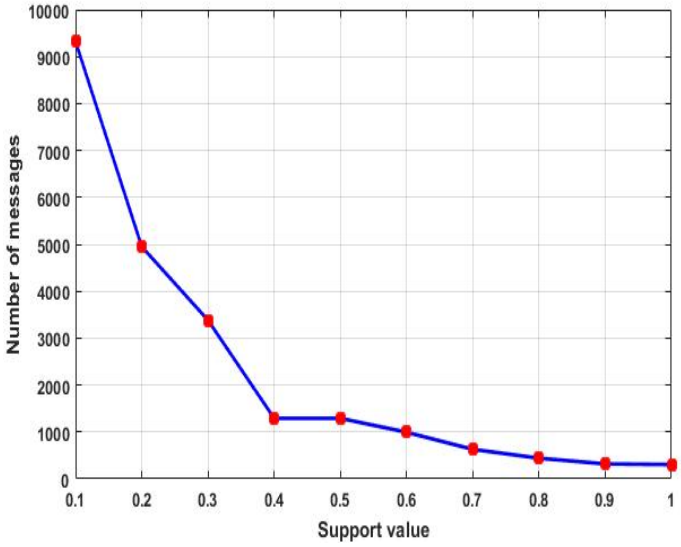

Fig. 1. Number of messages versus support values

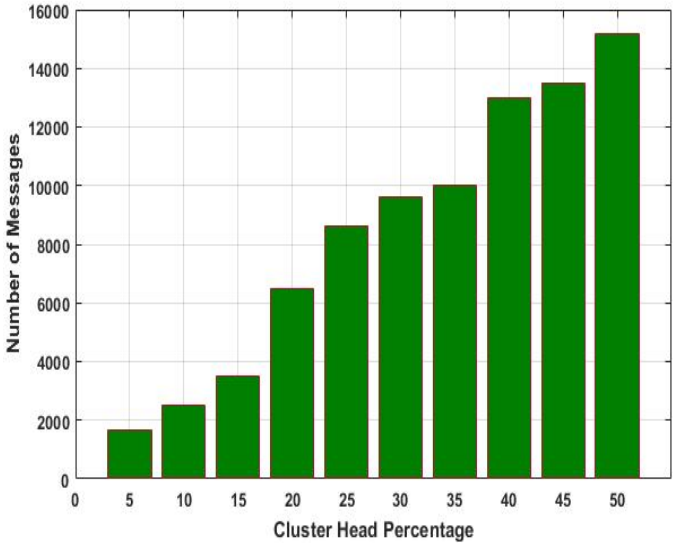

Fig. 2. Percentage of CHs versus number of messages

Cluster head percentage: In the second test, we use the same setting as in previous test except that the percentage of CHs varies from 5% to 50% with incremental of 5% and at each percentage the number of messages are computed. Fig. 2 shows the effect of the CH percentage on the number of messages. It is noted that as the percentage of cluster heads increases the number of messages increases because when the percentage of cluster heads increases, this will increase the messages to base station. Moreover, number of extracted rules increased as number of clusters increased which leads to increase the number of messages.

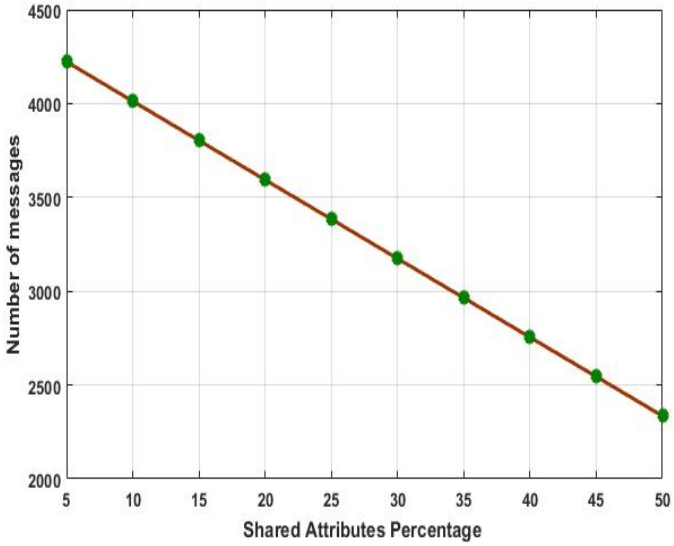

Fig. 3. Number of messages versus percentage of shared attributes

Percentage of shared attributed: In the third test, we use the same setting as in the first test except, the percentage of shared attributes varies from 5% to 50% with incremental of 5% and at each percentage the number of messages are computed. Fig. 3 shows the effect of the number of shared attributes on the number of exchanged messages. We can notice that as the percentage of shared attributes increases, the number of messages decreases because as the number of shared attributes increases, the percentage of unshared attributes decreases and so the number of messages needed for finding and controlling unshared attributes decrease.

VI. Conclusion

In this paper, we have proposed a new approach for mining the sensors data without moving these data to cluster heads or base station to achieve maximum performance in a WSN environment and keep the privacy of the sensor data. The proposed scheme maximizes local computations by utilizing the computing power at each sensor node and reduces the amount of the sensor data transferred in order to decrease the energy dissipation in communication, and thereby the

sensor network lifetime is maximized. Moreover, the proposed scheme includes a privacy-preserving technique to ensure the privacy of the sensor data by sending only a summary of the data between the cluster heads and its cluster members and between the cluster heads and the base station. In the future, we will conduct more experiments with different test metrics.

ACKNOWLEDGMENT

The authors gratefully acknowledge the Qassim University, represented by the Deanship of Scientific Research, on the material support for this research under the number (mcs-2018-1-14-S-4006) during the academic year 1440 AH /2019 AD.

REFERENCES

[1] J. Gama, P. P. Rodrigues, and L. Lopes, "Clustering distributed sensor data streams using local processing and reduced communication," Intelligent Data Analysis, vol. 15, no. 1, pp. 3–28, 2011.

[2] A. Boukerche and S. Samarah, "An efficient data extraction mechanism for mining association rules from wireless sensor networks," in Proceedings of the IEEE International Conference on Communications (ICC '07), pp. 3936–3941, June 2007.

[3] Y. Chi, H. Wang, P. S. Yu, and R. R. Muntz, "Moment: maintaining closed frequent itemsets over a stream sliding window," in Proceedings of the 4th IEEE International Conference on Data Mining (ICDM '04), pp. 59-66,November 2004.

[4] M. Deypir and M. H. Sadreddini, "EclatDS: an efficient sliding window based frequent pattern mining method for data streams," Intelligent Data Analysis, vol. 15, no. 4, pp. 571-587, 2011.

[5] A. Mahmood, K. Shi, and S. Khatoon, "Mining data generated by sensor networks: a survey," Information Technology Journal, vol. 11, pp. 1534-1543, 2012.

[6] J. Rabatel, S. Bringay, and P. Poncelet, "SO MAD: sensor mining for anomaly detection in railway data," in Advances in Data Mining. Applications andTheoretical Aspects, pp. 191-205, 2009.

[7] E. J. Spinosaa, A. P.D. L. F. deCarvalhoa, and J.Gamab, "Novelty detection with application to data streams," Intelligent Data Analysis, vol. 13, no. 3, pp. 405-422, 2009.

[8] P.A. Neves, J.J.P.C. Rodrigues, M. Chen, and A.V. Vasilakos, "A Multi-Channel Architecture for IPv6-Enabled Wireless Sensor and Actuator Networks Featuring PnP Support," J. Netw. Comput. Appl., vol. 37, pp. 12-24, Jan. 2014.

[9] J. M. L. P. Caldeira, J.J.P.C. Rodrigues, and P. Lorenz, "Towards Ubiquitous Mobility Solutions for Body Sensor Networks on HealthCare," IEEE Commun. Mag., vol. 50, no. 5, pp. 108-115, May 2012.

[10] L. Shu, Y. Zhang, G. Min, Y. Wang, and M. Hauswirth, "Cross-Layer Optimization on Data Gathering in Wireless Multimedia Sensor Networks within Expected Network Lifetime J. Univ. Comput. Sci., vol. 16, no. 10, pp. 1343-1367, 2009.

[11] C. Zhu, L. Shu, T. Hara, L. Wang, S. Nishio, and L.T. Yang, "A Survey on Communication and Data Management Issues in Mobile Sensor Networks," Wireless Commun. Mobile Comput., vol. 14, no. 1, pp. 19-36, Jan. 2014.

[12] S.R. Madden, M.J. Franklin, J.M. Hellerstein, and W. Hong, "TinyDB: An Aquisitional Query Processing System for Sensor Networks," ACM Trans. Database Syst., vol. 30, no. 1, pp. 122-173, Mar. 2005.

[13] Y. Yao and J. Gehrke, "Query Processing in Sensor Networks Proc. 1st Biennial Conf. Innovative Data Systems Research (CIDR) Asilomar, Pacific Grove, CA, USA, Jan.5-8, 2003.

[14] M. Umer, L. Kulik, and E. Tanin, "Optimizing Query Processing Using Selectivity-Awareness in Wireless Sensor Networks," J. Comput., Environ. Urban Syst., vol. 33, no. 2, pp. 79-89, 2009.

[15] L. Cheng, Y. Chen, C. Chen, J. Ma, L. Shu, A.V. Vasilakos, and N. Xiong, "Efficient Query-Based Data Collection for Mobile Wireless Monitoring Applications," Comput. J., vol. 53, no. 10, pp. 1643-1657, 2010.

[16] J.G. Pottie and J.W. Kaiser, "Wireless Integrated Network Sensors," Commun. ACM, vol. 43, no. 5, pp. 51-58, May 2000.

[17] R. Kacimi, "Energy Conservation Techniques for Wireless Sensor Networks," Ph.D. dissertation, Dept. Math., Info. and Telecom., INPT Univ., Toulouse, France, Sept. 2009.

[18] C. Dini, "Les Re seaux Capteurs Sans Fil Avec Access Sporadique Au Noeud-puits," Ph.D. dissertation, Info. Eng., Haute Alsace Univ., Mulhouse Cedex, France, Dec. 2010.

[19] E. Fasolo, M. Rossi, J. Widmer, and M. Zorzi, "In-Network Aggregation Techniques for Wireless Sensor Networks: A Survey," IEEE Wireless Commun., vol. 14, no. 2, pp. 70-87, Apr. 2007.

[20] C. Dini and P. Lorenz, "Primitive Operations for Prioritized Data Reduction in Wireless Sensor Network Nodes," in Proc. 4th ICSNC, Porto, Portugal, 2009, pp. 274-280.

[21] C. J. Debono and N. P. Borg, "The Implementation of an Adaptive Data Reduction Technique for Wireless Sensor Networks," in Proc. IEEE ISSPIT, Sarajevo, Bosnia, Dec. 16-19, 2008, pp. 402-406.

[22] L. Shu, J. Lloret, J.J.P.C. Rodrigues, and M. Chen, "Distributed Intelligence and Data Fusion for Sensor Systems," IET Commun., vol. 5, no. 12, pp. 1633-1636, Aug. 2011.

[23] A. Aziz, K. Singh,W. Osamy, and A. M. Khedr, Effective Algorithm for Optimizing Compressive Sensing in IoT and Periodic Monitoring Applications, Journal of Network and Computer Applications, vol. 126, pp. 12-28, 2019

[24] D. M. Omar, "ERPLBC: Energy Efficient Routing Protocol for Load Balanced Clustering in Wireless Sensor Networks, Ad Hoc & Sensor Wireless Networks, vol. 42, pp. 145-169, 2018.

[25] D. M. Omar, and A. M. Khedr, D. P. Agrawal Optimized Clustering Protocol for Balancing Energy in Wireless Sensor Networks, International Journal of Communication Networks and Information Security (IJCNIS) vol. 9, No. 3, pp. 367-375, December 2017.

[26] W. Osamy, A. M. Khedr, An algorithm for enhancing coverage and network lifetime in cluster-based Wireless Sensor Networks, International Journal of Communication Networks and Infor- mation Security (IJCNIS) Vol. 10, No. 1, pp. 1-9, April 2018.

[27] A. M. Khedr, and A. Attia, New Holes and Boundary Detection Algorithm for Heterogeneous Wireless Sensor Networks, International Journal of Communication Networks and Information Security (IJCNIS) vol. 10, No. 1, pp.163-169, April 2018.

[28] A. M. Khedr and D. M. Omar, SEP-CS: Effective Routing Protocol for Heterogeneous Wireless Sensor Networks, Ad Hoc & Sensor Wireless Networks, Vol. 26, pp. 211-232, 2015.

[29] R. Agrawal, R. Srikant, Fast algorithms for mining association rulesProceedings of the 20th International Conference Very Large Data Bases (VLDB '94)1994Citeseer487499

[30] R. J. Bayardo, Efficiently mining long patterns from databasesSIGMOD Record199827285932-s2.0-0032091573

[31] S. Brin, R. Motwani, and C., Silverstein, Beyond market baskets: generalizing association rules to correlationsSIGMOD Record19972622652762-s2.0-0031161999

[32] Cheung, W., Zaiane, O. R.Incremental mining of frequent patterns without candidate generation or support constraintProceedings of 7th International Database Engineering and Applications Symposium2003111116

[33] R. Agrawal, T. Imieliński, and A. Swami, Mining association rules between sets of items in large databasesProceeding of SIGMOD207216

[34] J. Han, J. Pei, Y. Yin, R. Mao, Mining frequent patterns without candidate generation: a frequent-pattern tree approach-Data Mining and Knowledge Discovery20048153872-s2.0-244244995210.1023/B:DAMI.0000005258.31418.83

[35] M. Halatchev, and L. Gruenwald, Estimating missing values in related sensor data streamsProceedings of the 11th International Conference on Management of Data (COMAD '05)2005

[36] N. Jiang, Discovering association rules in data streams based on closed pattern miningProceedings of the SIGMOD Workshop on Innovative Database Research2007.

[37] N. Jiang, L. Gruenwald, Estimating missing data in data streamsAdvances in Databases: Concepts, Systems and Applications20079819872-s2.0-38049151102

[38] K. Loo, I. Tong, B. Kao, Online algorithms for mining inter-stream associations from large sensor networksAdvances in Knowledge Discovery and Data Mining2005291302.

[39] G. S. Manku, R. Motwani, Approximate frequency counts over data streamsProceedings of the 28th International Conference on Very Large Data Bases2002346357

[40] S. K. Chong, S. Krishnaswamy, S. W. Loke, M. Gaber, Using association rules for energy conservation in wireless sensor networksProceedings of the 23rd Annual ACM Symposium on Applied Computing (SAC '08)March 20089719752-s2.0-5184912666910.1145/1363686.1363911

[41] S. K. Tanbeer, C. F. Ahmed, B. S. Jeong, Y. Lee, Efficient mining of association rules from wireless sensor networksProceedings of the 11th International Conference on Advanced Communication Technology (ICACT '09)February 20097197242-s2.0-67649882619

[42] A. Boukerche, S. A. Samarah, Novel algorithm for mining association rules in Wireless Ad Hoc Sensor NetworksIEEE Transactions on Parallel and Distributed Systems20081978658772-s2.0-5634909045810.1109/TPDS.2007.70789

[43] K. Romer, Distributed mining of spatio-temporal event patterns in sensor networksProceedings of the 1st Euro-American Workshop on Middleware for Sensor Networks (EAWMS '06) 2006.

[44] C. Wang, M. Chen, On the Complexity of Distributed Query Optimization. IEEE Transactions on Knowledge and Data Engineering, Vol. 8, No. 4, pp. 650–662, 1999.

[45] A. M. Khedr, Decomposable Algorithm for Computing k Nearest Neighbors across Partitioned Data, in: International Journal of Parallel, Emergent and Distributed Systems, vol. 31, no. 4, pp. Pages 334-353, 2016.

[46] A. M. Khedr and Raj Bhatnagar, New Algorithm for Clustering Distributed Data using k- means, Computing and Informatics, Vol. 33, pp. 1001-1022, 2014.

[47] F. A. Aderohunmu, J. D. Deng and M. K. Purvis, "A deterministic energy-efficient clustering protocol for wireless sensor networks," 2011 Seventh International Conference on Intelligent Sensors, Sensor Networks and Information Processing, Adelaide, SA, 2011, pp. 341-346. doi: 10.1109/ISSNIP.2011.6146592

[48] A. M. Khedr, Decomposable Algorithm for Computing k-Nearest Neighbors across Partitioned Data, in: International Journal of Parallel, Emergent and Distributed Systems, vol. 31, no. 4, pp. Pages 334-353, 2016.

[49] S. Dutt, S. Agrawal, and R. Vig. 2018. Cluster-Head Restricted Energy Efficient Protocol (CREEP) for Routing in Heterogeneous Wireless Sensor Networks. Wirel. Pers. Commun. 100, 4 (June 2018), 1477-1497. DOI: https://doi.org/10.1007/s11277-018-5649-x