Coverage-based Sensor Association Rules for Wireless Vehicular Ad hoc and Sensor Networks

Samer Samarah, Azzedine Boukerche, and Yonglin Ren
PARADISE Research Laboratory - University of Ottawa - Canada
{boukerch, ssamarah, yren009}@site.uottawa.ca

Abstract—Recently, Knowledge Discovery Process has proven to be a promising tool for extracting behavioral patterns regarding sensor nodes from wireless vehicular ad hoc and sensor networks. In this paper, we propose a new type of behavioral patterns, which we refer to as Coverage-based Rules, to discover the correlation among the set of locations monitored by the network. Coverage-base Rules is an extension for a recent proposed behavioral patterns named as Sensor Association Rules. However, in contrast to Sensor Association Rules, Coverage-based Rules have been designed specifically for sensor networks that guarantee a k-coverage property for the area under monitoring. The major application of Coverage-based Rules is to predict the location of future events. This feature might prove to be quite useful in vehicular ad hoc and sensor network based applications. To report about the efficiency of our proposed scheme, an extensive set of simulation experiments have been conducted to compare the performance of the network during the data preparation process for Coverage-based and Sensor Association Rules schemes.

I. INTRODUCTION

Advances in wireless technologies, vehicular ad hoc and sensor networks, and micro-electronic devices have led to the development of sensor nodes that are capable of sensing, processing, and transmitting data [12]. Sensor nodes are deployed in a certain geographical area, with the main functions of sensing the surrounding environment and sending the generated data to a central node, called the Sink, for further processing and analysis. In order to guarantee an acceptable level of quality for events’ delivery for vehicular ad hoc and sensor networks class of applications for instance, a new class of fast, reliable and fault tolerant protocols for WSN need to be developed. However, the distributed nature and the limited resources of sensor nodes (e.g., energy, communication and computation [12], [14]), as well as the unreliability of wireless communication increase the possibility of errors, lost messages, delays in data delivery and losses of functionality. These factors are potentially devastating to the performance and the overall Quality of Service (QoS) of WSNs [11].

Recently, Knowledge Discovery Process, a well known process in traditional database systems, used for extracting patterns from data [6], has shown to be a promising tool for improving WSN performance and its quality of services [10]. Knowledge Discovery in Wireless sensor network (KDW) has is used to extract two types of information (knowledge): (i) Patterns about the surrounding environment, extracted from the data reported by sensor nodes [19]. (ii) Behavioral patterns about sensor nodes, extracted from meta-data describing sensor behavior [1]. These kinds of patterns are referred to as sensors behavioral patterns.

Sensor Association Rules is a kind of behavioral pattern proposed in [1]; it aims to capture the temporal relations between sensor nodes, based on common intervals of their activities. An example of sensor rule is \( (s_1, s_2 \Rightarrow s_3, 90\%, \lambda) \), which translates: “if events from sensors \( s_1 \) and \( s_2 \) are received, then there is a 90% chance of receiving an event from sensor \( s_3 \) within \( \lambda \) units of time.” Formulating Sensor Association Rules entails: finding patterns of sensors that detect events within the same time interval; and the frequency with which the sensors detect an event within a certain time interval. The major impacts of Sensor Association Rules that benefit many applications are the ability of predicting the sources of future events, and the ability of identifying the sets of temporally correlated sensors.

In order to prepare the data needed for generating Sensor Association Rules, each sensor node should monitor its activity over time, and inform the Sink about the time slots in which events were detected (time is assumed to be divided into equal-sized slots). The set of time slot numbers in which the sensor has detected events within a given historical period is referred to as the Activity Set of the sensor node.

Sensor rules discover the correlation between all sensor nodes in the wireless sensor network, regardless of their locations. However, in most applications, a specific location may be covered by many sensors. Moreover, a k-coverage area, \( k > 1 \) is one of the desired Quality of Services (QoS) properties for WSNs [4]. In a k-coverage sensor network, all \( k \) nodes within a specific area are expected to detect the same event; as a result, all \( k \) sensor nodes will have the same activity sets during the process of profiling their behaviors when preparing the data needed for generating Sensor Association Patterns. Generating associations between all the sensor nodes, in a k-coverage networks, will result in a redundant information to the Sink that already known by the coverage property of the network.

In this paper, we propose a relaxation version of Sensor Association Rules that discovers the correlation between a set of locations (areas) rather than individual sensor nodes. We refer to the new proposed rules by Coverage-based Sensor Association Rules. The proposed rules will be based on partitioning the network into a set of areas, each cover by k-sensor nodes, \( k \geq 1 \). However, we do not assume that \( k \) is equal for all areas covered by this network, as it is difficult to guarantee a symmetric coverage, especially at the edges of the network [2].

This paper is organized as follows: Section II reviews some of the related works. Section III provides a formal definition
for Coverage-based Sensor Association Rules. Section IV includes simulation results prepared to evaluate the WSN’s performance during the data preparation process needed for generating Coverage-based Rules and Sensor Association Rules. Section V concludes the paper.

II. RELATED WORK

Knowledge Discovery process is by no means a trivial task; it requires a series of steps that include: Domain understanding, knowledge definition, data preparation and data mining [6]. Knowledge Discovery in Wireless sensor networks (KDW) is not, as of yet, a well-defined process. The ‘transformation’ from Knowledge Discovery in Databases (KDD) to Knowledge Discovery in Wireless sensor networks (KDW) warrants careful planning and refining for most of the techniques in KDD. Moreover, new techniques that are designed specifically for the knowledge discovery process in WSNs need to be envisioned [10].

Data Mining is an essential step in the knowledge discovery process that is concerned with extracting hidden knowledge from vast amounts of data using techniques inspired by different disciplines, such as databases, machine learning, artificial inelegant and statistics [13]. Recently, data mining techniques have been used to extract patterns regarding sensor data. These kind of patterns is usually used to gain insight about the phenomena under monitoring. These patterns can be also used to improve the performance of the network.

Association Rules algorithm is among the first data mining techniques that has been used to extract patterns from WSNs. Association Rules was first introduced by Argawal et al., [16] in 1993 to discover the correlations between objects in transactional databases, that appear in the same context. Market basket analysis is the main application of association rules; it uses them to predict purchase patterns of customers. Formulating predictions of such patterns help managers in making decisions about their businesses, such as appropriate items to go on sale, or placement of related items together (shelf management). Although Association Rules proposed for transactional database, it has been applied to different domains such as medical data [17].

Several works have been proposed for applying Association Rules to WSNs [18], [22], [19]. These works have targeted the data values of the sensor nodes; in other words, the sensors’ values are the main objects of the rules (i.e. temp =60 and light = "On" are some examples of the objects of these rules). Loo et al., [18] studies the problem of mining associations between sensors’ values in data streams generated from sensor nodes, in a particular wireless sensor network. Their technique depends on a data model that stores data reported from sensor nodes, and presenting it in a way that facilitates the adaption of lossy counting algorithm [20] that makes one pass analysis of the data. In this data model, sensors are assumed to take values from a finite discrete number of values. A quantization method is applied for the continuous values. The time is divided into equal-sized intervals, and snapshots from the sensors’ readings are taken whenever there is change to these readings. These snapshots are stored in a database in the form of contexts.

Mining spatial temporal event patterns is another attempt to link the problem of mining sensor data to the association mining problem, proposed by Kay Römer [19]. Römer’s approach took into consideration the distributed nature of wireless sensor networks, and proposed an in-network data mining technique, to discover the frequent patterns of events with certain spatial and temporal properties. In this approach, each sensor node is aware of the events that are within a certain distance from itself (this distance may be a Euclidean distance or number of hops). The sensor then collects these events and applies a mining algorithm to discover the pattern that satisfies a given parameters. Römer’s mining parameters approach includes minimum support S, minimum confidence C, maximum scope, and maximum history. Every node in the network collects the events from the neighbors within the maximum scope and keeps a history of their events for a duration of the maximum history. After that, each node applies a mining algorithm to discover the frequent patterns of the form:

\[ A_1 \land A_2 \land \cdots A_m \Rightarrow E[S,C]. \]  

which 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 the occurrence of event \( E \)" [19].

Halatchev et al., [22], proposed an association rule mining framework to tolerate the missed readings that resulted from the loss and corruption of messages while they being routed from sensor nodes to the Sink. Sensor readings are stream in nature, so applying an association mining algorithm like Apriori directly to the stream of data is not possible. This situation led authors to propose the Data Stream Association Rule Mining (DSARM) framework that transforms the Apriori algorithm [23] to make it applicable to the data stream received from sensor nodes.

In [1], the authors proposed Sensor Association Rules as an attempt to capture the temporal correlation between sensor nodes in a particular Wireless Sensor Network (WSN). The main difference of this approach from others is that the sensors themselves happen to be the main objects in the extracted rules, not the sensors’ readings. In order to prepare the data needed for generating Sensor Association Rules, each sensor node should monitor its activity over time, and inform the Sink about the time slots in which events were detected. This data is referred to by Sensor data activities. It is important to note that the data activities is meta-data that describes sensor behaviors (which differs from data reported by the sensor about the surrounding environment, i.e., sensors’ readings). Sensor data activities can be collected using existing routing protocols, or using one of several data gathering algorithms. However, none of these algorithms is designed to collect data for mining purpose and they do not consider the redundance existing between sensors’ data activity. These factors force the need for special data collecting techniques for mining purpose. In [21], three different mechanisms have been proposed for preparing the data needed for generating Sensor Association Rules. These
mechanisms are: (i) Direct Reporting; in this mechanism, sensor behavioral data is transferred to the Sink without any processing by the sensor node. (ii) Distributed Extraction; this mechanism is designed to put more computational load on the sensor nodes by equipping each sensor with additional storage space to store behavioral data in the historical period. Attaching a storage device to each sensor was previously impossible due to the high energy consumption caused by maintaining the storage device. However, recent advances in flash memory technology have changed this perspective, and studies have shown that energy consumption for maintaining a unit of data within a flash memory coupled with a sensor node is very low compared to the energy needed for transmitting that unit. In [8], the authors reported that a flash memory like NANAD from Toshiba, storing 28GB of data generated at a rate of 512 bytes per second, will decrease the life time of a NANAD from Toshiba, storing 28GB of data generated at a rate of 512 bytes per second, will decrease the life time of a sensor node designed to run for three years by only six weeks.

(iii) In-Network Reduction; in this mechanism, the number of messages needed to encapsulate the data is minimized by removing some of the redundancy presented in sensors’ data activity.

III. COVERAGE BASED RULES- FORMAL DEFINITION

In this section, we present a formal definition for Coverage-based Association Rules. We will use the same parameters used in defining Sensor Association Rules, which was presented in [1]. The parameters used include: slot size (λ), historical period (T_his) and minimum support. Time is divided into a set of time slots, of size λ each. Also, we assume that the network uses a buffering mechanism to profile network activities during the given historical period. In this buffering mechanism, each sensor maintains a buffer of size (T_his/λ). If the sensor detects an event within this time slot, a buffer entry corresponding to the current time slot is set. The network is divided into a set of locations, where each location is covered by k-sensors. Each location has a distinguished node known as the location manager. The location manager is responsible for collecting all the activity sets from other nodes within its coverage area at the end of the historical period. Recall that the activity set of a node s, \( AS(s) = \{l_1, l_2, ..., l_m\} \), is the set of time slot numbers in which the sensor detected events (i.e., the set of set entries in the sensor’s buffer).

Definition 3.1: Let \( L = \{l_1, l_2, ..., l_n\} \) be the set of locations in a particular sensor network, and let \( C(l_i) = \{s_1, s_2, ..., s_k\} \) be the set of sensors covering location \( l_i \). The activity set of \( l_i \), \( AS(l_i) \), is defined as the set produced by the union of all the activity sets of the nodes in \( C(l_i) \):

\[
AS(l_i) = \bigcup_{s_j \in C(l_i)} AS(s_j) \quad (2)
\]

Definition 3.1: A Location Database (DL) is defined as the set of locations covered by a particular sensor network, along with their activity sets.

Table I shows an example of a location database.

Definition 3.2: \( P = \{l_1, l_2, ..., l_m\} \), such that \( P \subseteq L \), is a pattern of locations.

### Table I

<table>
<thead>
<tr>
<th>Location</th>
<th>Activity set</th>
</tr>
</thead>
<tbody>
<tr>
<td>( l_1 )</td>
<td>{1, 3, 4, 6}</td>
</tr>
<tr>
<td>( l_2 )</td>
<td>{2, 4, 6}</td>
</tr>
<tr>
<td>( l_3 )</td>
<td>{1, 3, 5, 6}</td>
</tr>
<tr>
<td>( l_4 )</td>
<td>{2, 4, 5, 6}</td>
</tr>
</tbody>
</table>

Definition 3.3: The support of pattern \( P = \{l_1, l_2, ..., l_m\} \) in DL is defined by the cardinality of the set that is produced by intersecting all the activity sets of the locations in this pattern.

\[
\text{Support}(P) = \bigcap_{l_j \in P} |AS(l_j)| \quad (3)
\]

Definition 3.4: A pattern \( P \) is said to be frequent patterns if it support greater or equal a given minimum support.

Definition 3.5: Coverage-based Rule is defined by the implication \( P' \Rightarrow P'' \) where \( P' \subseteq L \), \( P'' \subseteq L \), and \( P' \cap P'' = \phi \).

Definition 3.6: The support of the rule \( (P' \Rightarrow P'') \) is defined by the support of the pattern \( (P' \cup P'') \) in DL, while the confidence of the rule is defined by:

\[
\text{Conf}(P' \Rightarrow P'') = \frac{\text{Support}(P' \cup P'')}{\text{Support}(P')}.
\]

Knowledge Discovery for Coverage-based Rules is the process of generating all the possible rules in the location database that meet an application’s pre-defined minimum support and confidence percentage.

An interesting application for Coverage-based Rules sees them predict the location of moving objects. For example, consider the network in Figure 1; this network is designed for detecting moving vehicles and reporting their locations to the Sink. Assuming that the Sink possesses a coverage rule that stipulates: \((E,B \Rightarrow A)\) that meets the application’s minimum support and confidence percentage, once it receives reporting messages from the location managers in location \( E \) and \( B \), it can easily predict that the next location of the moving vehicle will be \( A \). In this case, the Sink can inform the application about the next expected location of the moving object, a piece of information that proves helpful in many situations.

Generating Coverage-based Rules involve preparing the database that consists of a set of locations in the wireless sensor network and their activity sets. The key for preparing this database is to partition the network into a set of locations, and to identify the set of sensor nodes that cover each location. However, for the simulation purpose we will use a fixed partitioning made by the Sink. Recall that we are not proposing a new coverage-based clustering algorithm, as it is out of the scope of this paper and requires a deep analysis of its efficiency. We are going to provide a basic clustering scheme based on the sensing coverage of the sensor nodes in order...
to verify the basic concepts of our proposed Coverage-based Association Rules.

IV. PERFORMANCE EVALUATION

In this section, we present a comparison analysis for the performance of wireless sensor networks during the data preparation process for Coverage-based and Sensor Association Rules schemes. For simplicity, we will refer to the scheme that generates the association rules between all the sensor nodes as the "All-rules scheme." The metrics used to evaluate the performance of the network are: the number of messages needed to report the activity sets to the Sink, and the average energy consumptions per sensor node. The metrics are collected based on a simulator that has been built using Matlab version 7.4.0.287 [15]. The nodes are assumed to be deployed in a grid environment with a distance of 8 meters between sensor nodes and a sensing range up to 16 meters. Events generation, by each sensor node, is assumed to be uniformly distributed over the possible number of epochs. Sensor nodes covering the same area have the same activity sets.

Although Coverage-based Rules can be implemented on top of any of the coverage-based clustering algorithms, like those introduced in [2], [3], we have implemented it on top of fixed network partitions, where the location managers have been decided by the Sink node, based on the network topology. The number of location managers depends on the sensing range and the density of the network. If the network is dense enough, and the nodes have long sensing range, then the number of partitions is equal to N/k, where N is the number of sensors in the network and k is the required coverage property. In reality, however, we may have more than this number, as it is hard to guarantee full k-coverage networks, especially at the borders of the network [2]. To cluster the network, we assume that all the nodes have the same sensing range (R) and transmission range (T), where T >= R. N(x) is the set of x’s neighbors within radius r, where r = (R/2). Each location manager sends a joint message to all nodes in its neighbor set. A node waits until it receives all the possible joint messages before it joins the closet cluster. The partitioning is done in such a way that all the areas in the network are covered. The coverage property ranges between 2-4 nodes at the border of the network, and 6 nodes at the central of the network. Each node uses its full sensing coverage for detecting events. Events outside a radius of (R/2) of the location manager (i.e., r)-8 m in our setting will be discarded. Figure 2 shows the sensing range of the sensor nodes.

For the All-rules scheme, we have used the Distributed Extraction mechanism, which was introduced in [1], to prepare the data needed for generating the association rules. In both schemes, the nodes use the buffering mechanism to profile their activities during the given historical period. At the end of the historical period, defined by the application, the node sends its Activity Set to the Sink (in case of All-rules scheme), or to the location manager (in case of Coverage-based scheme) if the cardinality of this set is greater or equal the given minimum support. To route the activity sets to the Sink, a multi-hops tree has been constructed based on the minimum distance to the Sink. For energy consumption, we have used the first order radio model introduced in [7]. Equation 4 shows the energy consumption for sending a k bit message across distance d. E_{elec} is the energy consumption used to run the transmitter and the receiver, which is 50 nJ/bit. E_{amp} = 100 pJ/bit/m^2 is the energy consumption for the transmitter’s amplifier. In our experiments, we have assumed that k=480 bits and d=15m. Equation 5 shows the energy consumption of turning the receiver circuit. Also, there is an extra cost incurred for maintaining the storage device. We have assumed a Toshiba 16MB NAND flash memory that costs 0.017 uJ to read, write, and erase a byte of data [8], [9]. Table II summarizes the parameters used in our experiments.

\[ E_{TX}(k, d) = E_{elec} \times k + E_{amp} \times k \times d^2 \]  
\[ E_{RX}(k) = E_{elec} \times k \]  

TABLE II

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of nodes</td>
<td>500, 1000</td>
</tr>
<tr>
<td>Historical period</td>
<td>1-10 days</td>
</tr>
<tr>
<td>Flash memory size</td>
<td>16 MB</td>
</tr>
<tr>
<td>Minimum support</td>
<td>10% - 90%</td>
</tr>
<tr>
<td>Slot size</td>
<td>30 sec</td>
</tr>
<tr>
<td>Message size</td>
<td>60 Byte</td>
</tr>
<tr>
<td>Read, write, and erase from flash</td>
<td>0.0017 \mu J/Byte</td>
</tr>
<tr>
<td>Transmission and receive energy</td>
<td>50 \mu J/m</td>
</tr>
<tr>
<td>Transmitter’s amplifier</td>
<td>100 pJ/bit/m^2</td>
</tr>
<tr>
<td>Grid size</td>
<td>(185 m \times 185 m, 248m \times 248m)</td>
</tr>
<tr>
<td>Sensing range</td>
<td>16 m</td>
</tr>
<tr>
<td>Transmission range</td>
<td>16 m</td>
</tr>
<tr>
<td>Distance between sensors</td>
<td>8 m</td>
</tr>
<tr>
<td>Coverage property</td>
<td>1-6 nodes</td>
</tr>
</tbody>
</table>

Different experiments were conducted to collect the desired metrics. Figure 3 and Figure 4 show the total number of messages needed to report the nodes’ activity sets, for the All-rules and the Coverage-based Rules schemes, for support values ranging between 10% to 90%, and number of nodes equal to 500 and 1000, respectively. Results show that Coverage-based Rules requires fewer messages to extract data than the All-rules scheme; in numbers, and for the given simulation parameters, Coverage-based Rules requires 7%-10% of the number of messages needed for preparing the behavioral data needed for All-rules. This percentage, however, depends on the coverage property of the network. The higher the degree of clustering, the fewer is the volume of data to be routed to the Sink. In the worst case (k = 1), the number of messages for the Coverage-based Rules will be equal to that needed to prepare the data for the All-rules scheme, in addition to a few number of control messages needed in the attempt for clustering the network. If Coverage-based Rules is built on top of an already coverage-based clustering algorithm, then this number can be neglected.

Another interesting metric about network performance during the process of preparing the data for All-rules and Coverage-based Rules is the average energy consumption per node. We considered two main sources for energy consumption: that needed for maintaining the buffer storage at each node (i.e., the energy consumption for read, write and erase a buffer entry), and the energy exhausted by message transmission. We have compared the average energy consumption...
per node of network topologies of 500 and 100 nodes, and historical period ranges of 1 to 10 days. Figures 5, 6 and 7 show the energy consumption per node at minimum support values of 0, 50% and 90% respectively. All figures show that the energy consumption per node for the Coverage-based Rules is less than the average energy consumption using the All-rules scheme; in Coverage-based Rules, each node requires anywhere between 50% to 55% of the energy exhausted by a node in the All-rules scheme. The percentages will vary if the number of messages changed. However, there is extra energy consumption to be accounted for in case of Coverage-based Rules, due to the transmission of activity sets from sensor nodes to the location manager.

V. CONCLUSION

Coverage-based Association Rules is the Knowledge Discovery technique, designed specifically for wireless networks, that guarantees a k-coverage property for the network. In contrast to Sensor Association Rules, Coverage-based Rules discovers the correlation between the set of locations in the network, instead of the correlation between all nodes. The activity set of each location is prepared from the activity sets of the sensor nodes covering the area. In order to evaluate the performance of our proposed scheme, we have presented several set of experiments. Our results indicate clearly that our proposed coverage-based Rules require about 7%-10% of the number of messages used by the All-rules scheme as discussed earlier in Section 4, and in terms of average energy consumption, we have observed that each sensor node in our coverage-based Rules scheme requires about 50-55% of the energy exhausted by a node in the All-rules scheme.

REFERENCES


