

A Tree Structure for Mining Behavioral Patterns from Wireless

Nandigam Srinivas¹, Mr. Simma Seshagiri²

¹Final M.Tech Student, ²Associate professor

^{1,2}Department of CSE, Sarada Institute of Science,

^{1,2}Technology and Management (SISTAM), Srikakulam, Andhra Pradesh, India

ABSTRACT

Now a day's wireless sensor network interesting research area for discovering behavioral patterns wireless sensor network can be used for predicting the source of future events. By knowing the source of future event, we can detect the faulty nodes easily from the network. Behavioral patterns also can identify a set of temporally correlated sensors. This knowledge can be helpful to overcome the undesirable effects (e.g., missed reading) of the unreliable wireless communications. It may be also useful in resource management process by deciding which nodes can be switched safely to a sleep mode without affecting the coverage of the network. Association rule mining is the one of the most useful technique for finding behavioral patterns from wireless sensor network. Data mining techniques have recent years received a great deal of attention to extract interesting behavioral patterns from sensors data stream. One of the techniques for data mining is tree structure for mining behavioral patterns from wireless sensor network. By implementing the tree structure will face the problem of time taking for finding frequent patterns. By overcome that problem we are implementing associated correlated bit vector matrix for finding behavioral patterns of nodes in a wireless sensor network. By implementing this concept

1. INTRODUCTION

Wireless Sensor Network generates a large amount of data in the form of data stream and mining these streams to extract useful knowledge is a highly challenging task. In literature study, existing mechanism use sensor association rules measured in terms of frequency of patterns occurrence. Among the enormous number of rules generated, most of those are not valuable to reproduce true association among data objects. Moreover, mining associated sensor patterns from sensor stream data is essential for real-time applications, but it is not addressed in literature papers. In this proposed work, a new type of sensor behavioural pattern called associated sensor patterns to capture substantial temporal correlations in sensor data simultaneously is introduced to address the above-said problem. In this paper we are proposed an efficient associated correlated bit vector matrix for find the frequent item sets and associate correlated frequent pattern sets. The Data Mining in WSN are used to extract useful data from the huge amount of unwanted dataset. The need of mining to get knowledgeable data and discovers the behavioural patterns. As there are many Association techniques in data mining to find out the Frequent Patterns as per . The Association rule can apply on static data and stream data. The frequent patterns are those items, Sequences or substructure which reprise from the available dataset by providing the user specified frequencies. Whenever you want to find out the frequently occurred data apply association rules which will find out the frequent patterns from the dataset. Mining play main role to mine frequent item set in many data mining tasks. Over data streams, the frequent item set mining is mine the approximation set of frequent item sets in transaction with given support and threshold. It should support the flexible determine between mining accuracy and processing time. When the user-specified minimum support threshold is small, it should be time efficient. To propose an efficient algorithm the objective is generates frequent patterns in a very less time.

Frequent patterns are very meaningful in data streams such as in network monitoring, frequent patterns relate an

indicator for network attack to excessive traffic. In sales transactions, frequent patterns correspond to the top selling products with their relationships in a market. If we consider that the data stream consist of transactions, each items being a set of items, then the problem definition of mining frequent patterns can be written as given a set of transaction and finds all patterns with frequency above a threshold. Wireless sensor networks (WSNs) are successfully deployed in diverse monitoring and detection applications. In these applications, WSNs generate a large amount of data in the form of streams. Such data stream from WSN can be mined to extract knowledge in real time about the sensed environment (e mining certain behaviors and the network itself, and this presents new challenges for data mining techniques. Data mining techniques, which are well established in the traditional database systems, have recently received a great deal of attention as promising tools to extract interesting knowledge from sensor data streams (SDSS). Using knowledge discovery in WSN, one particular interest is to find behavioral patterns of sensor nodes, which are evolved from meta-data describing sensor behaviors. Data mining techniques, well established in the traditional database systems, recently became a popular tool in extracting interesting knowledge from sensor data streams (SDSS). Using knowledge discovery in WSNs, one particular interest is to find behavioral patterns of sensor nodes evolved from meta-data describing sensor behaviors. The application of fine grain monitoring of physical environments can be highly benefitted from discovering behavioral patterns (i.e., associated patterns) in WSNs. These behavioral patterns can also be used to predict the cause of future events which is used to detect faulty nodes, if any, in the network. For example, possibility of a node failure can be identified using behavioral pattern mining by predicting the occurrence of an event from a particular node, but no such event reported in subsequent iteration. As behavioral patterns reveal a chain of related events, source of the next event can be identified. For e.g. in an industry, fault in a particular process may trigger fault in other processes. In addition, behavioral patterns can also use to identify a set of

temporally correlated sensors, thus improving operational aspects in WSNs. wireless sensor network is a collection of nodes organized into a cooperative network. It consists of spatially distributed autonomous sensors to monitor physical or environmental conditions, such as temperature, sound, pressure, etc. and to cooperatively pass their data through the network to a main location predefined as central node called sink in multi-hop mode of transmission. Not only the unreliable wireless communication, the node of wireless sensor networks have to work with limited resources such as limited energy, limited processing capacity, limited storage, limited memory, limited communication capacity, etc. So, wireless sensor networks suffers from lot of problems such as lost messages, delay in data delivery, loss of data, data redundancy, etc. resulting in poor quality of service.

EXISTING SYSTEM

However, association rule mining with real datasets is not so simple. If the minimum support threshold is high, then we can get high value knowledge. On the other hand, when the minimum support threshold is low, an extremely large number of association rules will be generated, most of the are non-informative. The valid correlation relationships among data objects are buried deep among a large pile of useless rules. Additionally, association rules mining are not able to discover such kind of patterns where the event detected by sensor s_1 can increase the likelihood of the event detect by s_2 . Such kind of patterns are both associated and correlated. To overcome this difficulty, here we combine association and correlation in the mining process to find the associated correlated patterns. To generate associated-correlated patterns that have a certain frequency (support) it is required to generate all the patterns present in the data, i.e., frequent patterns. Once the frequent patterns are determined, the process of generating the associated-correlated patterns is then straight forward. Therefore, devising an efficient algorithm to mine frequent pattern with high speed in large-scale sensor data has been the real challenge.

PROPOSED SYSTEM

Mining play main role to mine frequent item set in many data mining tasks. Over data streams, the frequent item set mining is mine the approximation set of frequent item sets in transaction with given support and threshold. It should support the flexible determine between mining accuracy and processing time. When the user specified minimum support threshold is small, it should be time efficient. To propose an efficient algorithm the objective is generates frequent patterns in a very less time. Frequent patterns are very meaningful in data streams such as in network monitoring, frequent patterns relate an indicator for network attack to excessive traffic. In sales transactions, frequent patterns correspond to the top selling products with their relationships in a market. If we consider that the data stream consist of transactions, each items being a set of items, then the problem definition of mining frequent patterns can be written as given a set of transaction and finds all patterns with frequency above a threshold. In this paper we are proposed an efficient correlated association rule mining for mining behavioral patterns from wireless sensor network. For mining association correlated patterns can be done by performing the two steps. In the first step we are finding frequent patterns of wireless sensor networks and second step is to test whether they are associated correlated

patterns or not based on the confidence of each pattern in a transaction dataset. By performing those two operations we are implementing an efficient correlated bit vector matrix for finding behavioral patterns from a wireless sensor network. The implementation procedure of associated correlated bit vector matrix is as follows.

Associated Correlated Bit Vector Matrix

Frequent Pattern mining techniques find the candidates and frequent patterns generated. In frequent pattern mining techniques for finding frequent patterns contained two problems they are, many times scanned the database and more complex candidate generation process.

To find the frequent patterns with single scan of database, we propose a technique associated correlated bit vector matrix which is used to generate associated patterns. The generation frequent patterns of sensor stream of data is as follows. Generation of Bit Vector Matrix:

In this module we can retrieve the transactional data set of sensor items from the data base. Take the each transaction and generate bit vector matrix. The implementation of bit vector matrix is as follows.

1. Read each transaction from the data base (D) and get each item of sensor node id S_i .
2. Read all the individual items of sensor nodes until the length of all transactional dataset is completed.
3. After completion of reading process we can sort the all node ids.
4. Find all frequent length of item sets (T_i) from the data base D

If T_i is not null

For each transaction (T_i) from database

For each item (I_i) in database D

If item (I_i) contains Transaction Item sets (T_i)

BV = 1

Else

BV=0

End for.

End for.

End if.

If all the maximum frequent item sets are empty, the maximum length of frequent item set is one.

Input: the bit vector matrix, Minimum support value

Output: Maximum Frequent patterns of sensor nodes

Process:

For each column in the bit vector matrix

Calculate number of value one in the current row

End for

Return max[n]

Sort (Max[n])

For each one in the max[n]

Calculate number of columns with the same number of ones

If number > minimum support value

Generate maximum number of candidate item sets from transaction

For each item set in candidate item sets

Calculate support (item set)
 If (support (item set)>minimum support count)
 Item set is frequent
 18
 End if
 End for
 End if
 If maximum item sets is not null
 Break;
 End if
 End for.

- > Any activity
- > Individual component of the system
- > And how they can interact with the other components
- > How the system will run
- > How entities interact with others (components and interfaces)
- > External user interface

By calculating confidence of two sensor, such as s1 , s2is as follows.

$$p(s1,s2) = P(s1,s2)-P(s1)P(s2)/P(s1,s2)+P(s1)P(s2)$$

Suppose we are calculating more than two nodes of confidence we are using the following formula.

$$P=P(s1,s2....sn)-P(s1)P(s2)....P(sn)/P(s1,s2....sn)+P(s1)P(s2)....P(sn)$$

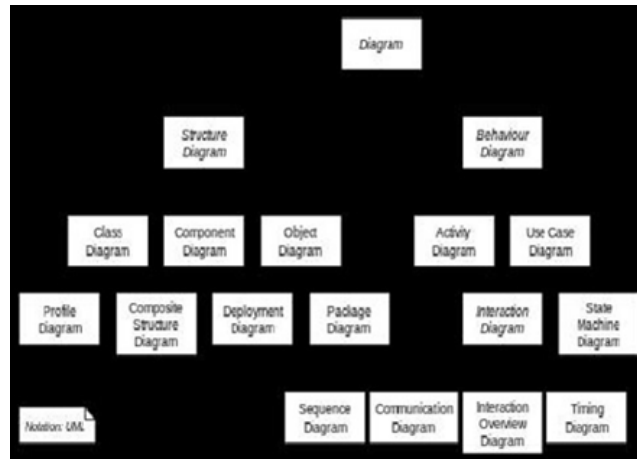


Fig 1 Block Diagram of various UML diagrams

2. REQUIREMENT ANALYSIS

Hardware Requirements

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatible, and sometimes incompatible hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

Hardware Requirements for Present Project

1. VDU: Monitor/ LCD TFT / Projector
2. Input Devices: Keyboard and Mouse
3. RAM: 512 MB
4. Processor: P4 or above
5. Storage: 10 to 100 MB of HDD space.

Software Requirements

Software Requirements deal with defining software resource requirements and pre-requisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or pre-requisites are generally not included in the software installation package and need to be installed separately before the software is installed.

Software Requirements for Present Project:

1. Operating System: Any Operating System
2. Run-Time: OS Compatible JVM

3. SYSTEM DESIGN

Unified Modeling Language (UML) :

The Unified Modeling Language (UML) is a general-purpose, developmental, modeling language in the field of that is intended to provide a standard way to visualize the design of a system. The Unified Modeling Language (UML) offers a way to visualize a system's architectural blueprints in a diagram (see image), including elements such as:

Use Case Diagram

Use Case Relationships

Interaction among actors is not shown on the use case diagram. If this interaction is essential to a coherent description of the desired behavior, perhaps the system or use case boundaries should be reexamined. Alternatively, interaction among actors can be part of the assumptions used in the use case.

Actor Generalization

One popular relationship between Actors is Generalization/Specialization. This is useful in defining overlapping roles between actors. The notation is a solid line ending in a hollow triangle drawn from the specialized to the more general actor.

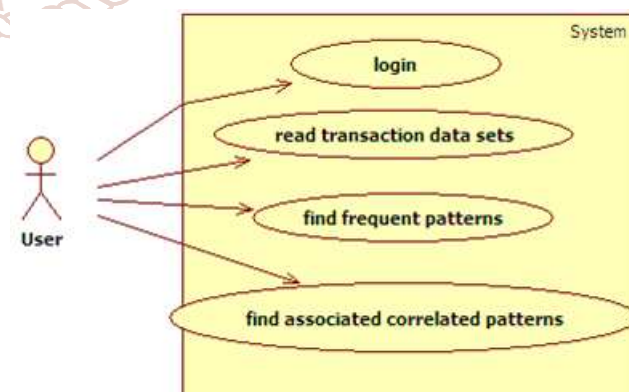
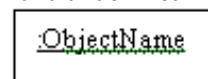


Fig 2 Elements of Collaboration diagram

A Collaboration diagram consists of the following elements:

Object:

The objects interacting with each other in the system. Depicted by a rectangle with the name of the object in it, preceded by a colon and underlined.



Relation/Association:

A link connecting the associated objects. Qualifiers can be placed on either end of the association to depict cardinality.



Messages:

An arrow pointing from the commencing object to the destination object shows the interaction between the objects. The number represents the order/sequence of this interaction.

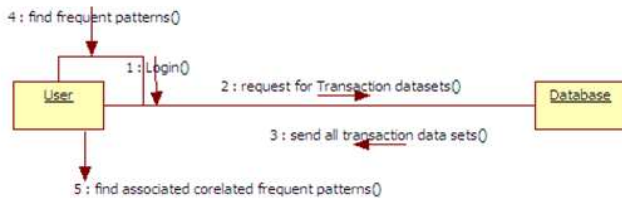


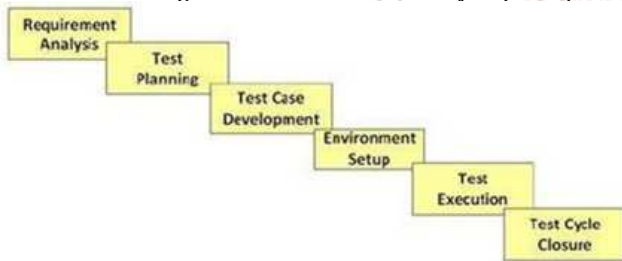
Fig 3 collaboration diagram

4. TESTING

Software testing can also be stated as the process of validating and verifying that a software program/application/ product:

1. meets the business and technical requirements that guided its design and development;
2. Works as expected; and
3. Can be implemented with the same characteristics.

The different stages in Software Test Life Cycle(STLC)



Test Planning

This phase is also called Test Strategy phase. Typically, in this stage, a Senior QA manager will determine effort and cost estimates for the project and would prepare and finalize the Test Plan

Activities:

- Preparation of test plan/strategy document for various types of Testing
- Test tool selection
- Test effort estimation
- Resource planning and determining roles and responsibilities.
- Training requirement

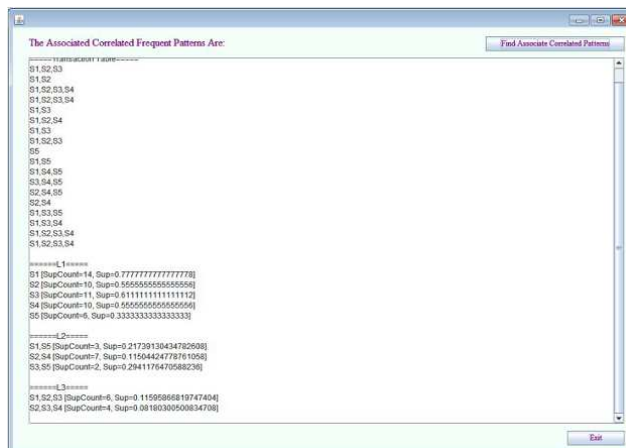
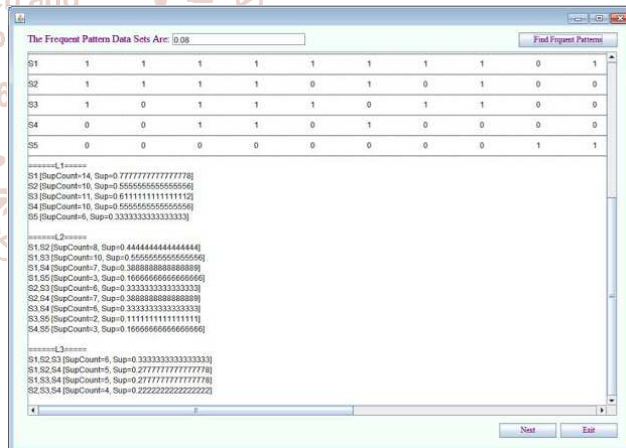
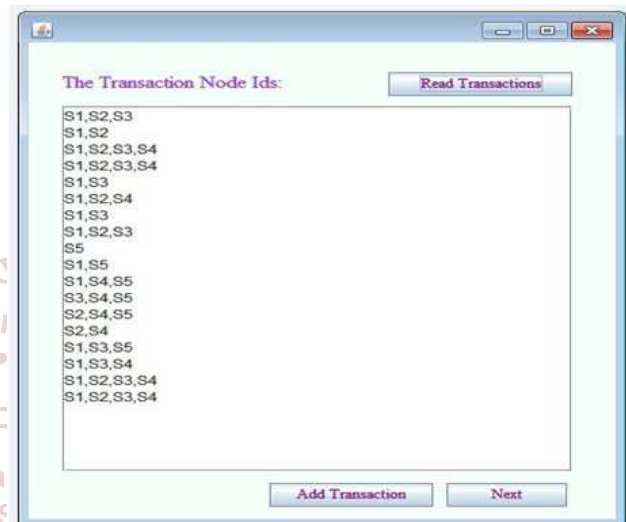
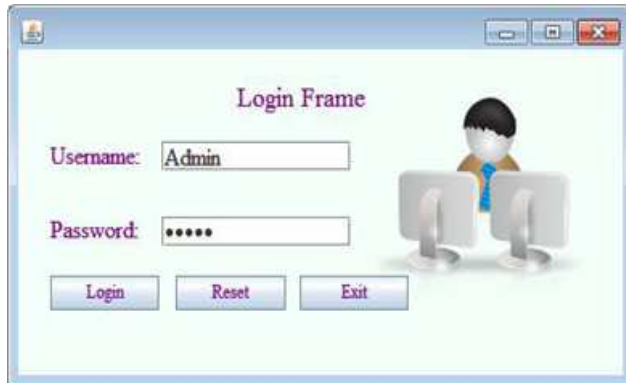
Test Case Development

This phase involves creation, verification and rework of test cases & test scripts. Test data, is identified/created and is reviewed and then reworked as well.

Activities

- Create test cases, automation scripts (if applicable)
- Review and baseline test cases and scripts
- Create test data (If Test Environment is available)

5. RESULTS



6. CONCLUSION

In this paper we are proposed an efficient association rule mining finding associated correlated frequent behavioral patterns from wireless sensor network. Our proposed associated correlated bit vector matrix for mining behavioral frequent patterns of wireless sensor network data. By implementing this process we can scan the entire data once and mine many properties is suitable for interactive mining. An extensive analysis of associated correlated bit vector matrix is finding associated frequent patterns mining and out performs the existing algorithm based on execution time and memory usage.

7. REFERENCES

- [1] M. M. Rashid, I. Gondal and J. Kamruzzaman, (2013) Mining associated sensor patterns for data stream of wireless sensor networks, in Proc. 8th ACM Workshop Perform. Monitoring Meas. Heterogeneous Wireless Wired Netw., , pp. 91-98
- [2] Jiinlong, XuConglfu, CbenWeidong, Pan Yunhe," Survey of the Study on Frequent Pattern Mining in Data Streams", 2004 IEEE International Conference on Systems, Man and Cybernetics.
- [3] R. Agrawal and R. Srikant, "Fast algorithms for Mining Association Rules", 20th International Conference on Very Large Data Base, pp. 487-499, May1994.
- [4] Md. Mamunur Rashid, IqbalGondal and JoarderKamruzzaman, "Share-Frequent Sensor Patterns Mining from Wireless Sensor Network Data", IEEE Transaction Parallel Distribution System, 2014.
- [5] Imielienskin T. and Swami A. Agrawal R., "Mining Association Rules Between set of items in large databases," in Management of Data, 1993, p. 9.
- [6] H. Mannila, R. Srikant, H. Toivonen, and A. Inkeri R. Agrawal, "Fast Discovery of Association Rules," in Advances in Knowledge Discovery and Data Mining, 1996, pp. 307-328.
- [7] M. Chen, and P.S. Yu J.S. Park, "An Effective Hash Based Algorithm for Mining Association Rules," in ACM SIGMOD Int'l Conf. Management of Data, May, 1995.
- [8] R. Motwani, J.D. Ullman, and S. Tsur S. Brin, "Dynamic Itemset Counting And Implication Rules For Market Basket Data," ACM SIGMOD, International Conference on Management of Data, vol. 26, no. 2, pp. 55-264, 1997.

