



**Developing Parallel Technique for the Internet of Things
Devices by using local Map Reduce at the sink node.
(تطوير تقنيات موازية لأجهزة أنترنت الأشياء باستخدام متعقب الوظيفة المحلي في
عقدة التجميع)**

Prepared by

Muna George Oweis

Supervised by

Dr. Sadeq AlHamouz

**A Thesis Submitted in Partial Fulfillment of the Requirements for the
Master Degree in Computer Science**

Department of Computer Science

Faculty of Information Technology

Middle East University

Amman, Jordan

May, 2016

Authorization

I, Muna George Oweis, Authorize the Middle East University to supply a copy of my thesis to libraries, establishment or individuals.

Name: Muna George Oweis





Date: 24/5/2016

Signature:



Examination Committee Decision

This is to certify that the thesis entitled "Developing Parallel Technique for the Internet of Things Devices by using local MapReduce at the sink node." was successfully defended and approved on 24/5/2016.

Examination Committee Members	Signature
(Supervisor) Dr. Sadeq AlHamouz Computer science department College of Information Technology Middle East University	
(Co-Supervisor) Dr. Maamoun Ahmed Head of Computer Science Department College of Information Technology Aqaba University of Technology	
(Head of the Committee and Internal Committee Members) Dr. Abdelrahman Abuarqoub Head of computer science department College of Information Technology Middle East University	
(External Committee Members) Prof. Mohammed Fendi Ababneh Computer science department College of Information Technology Balqa Applied University	

Acknowledgments

First and foremost, my thanks are hereby extended to God, then to my supervisors: Dr. Sadeq AlHamouz and Dr. Maamoun Ahmed for their supportive and helpful supervision, for assisting me in every step of the project, and for providing important information and skill set which was very important for the successful implementation of the thesis project. I would like to extend my thanks to the Information Technology Faculty members at the Middle East University. Further thanks are extended to everyone who helped me develop my understanding of the various nuances of the project and for everyone who believes that the knowledge is a right for everyone.

إهداء

إلى مَنْ هُمْ فِي الْحَيَاةِ حَيَاةً إِلَيْكُمْ يَنْحَنِي الْحَرْفُ حُبّاً وَامْتِنَانٍ
إلى مَنْ حَصَدُوا الْأَشْوَكَ مِنْ دُرِّي لِيَتَمَهَّدُوا لِي طَرِيقَ الْعِلْمِ وَالنَّجَاحِ إِلَى مَعْنَى الْحَبِّ وَالْحَنَانِ
وَالْتَفَانِي إِلَى أَعْلَى الْحَبَائِبِ

أُمِّي وَأَبِي

إلى مَنْ نَقَشَ مَعِيَ حُرُوفَ نَجَاحِي حَرْفاً بِحَرْفٍ رَفِيقٌ دُرِّي فِي رَحْلَتِي إِلَى مَنْ كَانَ
أُسْتَاذِي وَمُعَلِّمِي وَسَنَدِي

زَوْجِي نُور

نَبْضُ قَلْبِي وَقَلَمِي

إلى الْعَيْنَيْنِ اللَّتَيْنِ أَسْتَمْدُ مِنْهُمَا الْقُوَّةَ وَالِاسْتِمْرَارَ إِلَى مَنْ صَبَرُوا مَعِيَ عَلَى الْمَشَقَّةِ وَالتَّعَبِ أَبْنَائِي

عَيْسَى وَوَاتِل

أَنْتُمْ أَمَلِي

إلى الشَّمْعِ الْوَحِيدِ الَّتِي تَنِيرُ لِي طَرِيقِي إِلَى مَنْ كَانُوا عَوْنِي فِي الْحَيَاةِ إِخْوَتِي شَادِي وَوَسِيمِ وَوَاتِل

إلى الْجَوْهَرَةِ الْمُنِيرَةِ أُخْتِي الْغَالِيَةِ أُرُوِي

إلى عَائِلَتِي وَاصْدِقَائِي وَكُلِّ مَنْ وَقَفَ بِجَانِبِي وَكَانَ لَهُمُ الْفَضْلَ فِيمَا وَصَلْتُ إِلَيْهِ مِنْ نَجَاحٍ

إلى أَسَاتِدَتِي الْكَرَامِ كُنْتُمْ خَيْرَ مَنْ تُشَدُّ لَهُ الرِّحَالُ

إلى كُلِّ مَنْ يَبْحَثُ عَنِ الْعِلْمِ بَيْنَ ثَنَائِي هَذِهِ الْوَرِيقَاتِ

أَهْدِي جَهْدِي الْمَتَوَاضِعَ

Table of contents

Title.....	I
Authorization	II
Thesis Committee Decision	III
Acknowledgments	IV
Dedication.....	V
TABLE OF CONTENTS	VI
List of Figures	IX
List of Tables	X
List of Abbreviations	XI
List of Published Papers:	XII
Abstract.....	XIII
ملخص.....	XIV
Chapter One: Introduction	2.
1.1. Overview	2
1.2. Problem Statement	3
1.3. Research Question.....	4
1.4. Research Objectives	4
1.5. Research Motivation and Significance.....	5
1.6. Thesis Outline	6
2 Chapter two: Theoretical Framework and Literature Review	8
2. 1 Introduction	8
2. 2 Big Data Concept	8
2. 3 Big Data Tools	10
2. 4 Association Rules.....	13

2.1.	Internet of Things (IoT).....	15
2.1.1.	Internet of Things Hardware	15
2.1.2.	Wireless Sensor Network (WSN)	16
2.5	Literature Review	19
2.6	Related Work Overview	20
	Summary	28
3.	Chapter Three: Methodology	31
3.1.	Introduction	31
3.2.	System Model.....	32
3.1	Dataset Collection	33
3.2	Software Used	33
3.3	Apply MapReduce for IoT (MRIoT)	33
3.4	Throughput and Energy Consumption	37
4.	Chapter Four: Results and Discussion	40
4.1.	Introduction	40
4.2.	Assumptions	40
4.3.	Sensor Reading Results	41
4.3.1.	Key and Value.....	41
4.3.2.	System Parameters	43
4.3.3.	All Rules	44
4.3.4.	Accepted Rules	45
4.3.5.	Case Study 1 (TH = 0.001)	46
4.3.6.	Case Study 2 (TH = 0.1)	47
4.4.	Throughput and Energy Consumption Results	47
4.5.	Performance Evaluation	49

5. Chapter Five: Conclusions and future works.....	52
List of References	54
Appendix : Rules and MapReduce work	62

List of Figures

FIGURE 2-1: BIG DATA CHARACTERISTICS.....	9
FIGURE 2-2: BIG DATA TOOLS.	10
FIGURE 2-3: BASIC STRUCTURE OF THE MAPREDUCE PROCESS (NIU, Z., ET AL., 2012)	12
FIGURE 2-4: WSN ARCHITECTURE (IIT KHARAGPUR., 2015).....	16
FIGURE 2-5: ILLUSTRATION OF THE WSN APPLICATION (LIBELIUM., 2015)	18
FIGURE 2-6: THE WSN STRUCTURE (PAIK ET AL 2014)	18
FIGURE 3-1: METHODOLOGY DESIGN FOR MRIOT	31
FIGURE 3-2: MAPREDUCE SYSTEM MODEL FOR MRIOT	32
FIGURE 3-3: FLOW WORK OF THE MRIOT.....	34
FIGURE 4-1: KEY-VALUE RESULTS (KEY).....	41
FIGURE 4-2: KEY-VALUE RESULTS (VALUE).....	41
FIGURE 4-3: KEY FOR TWO VALUES	42
FIGURE 4-4:KEY FOR MULTI VALUES	42
FIGURE 4-5: ALL KEYS AND VALUES.....	42
FIGURE 4-6:ALL DATA RESULTS	44
FIGURE 4-7:CONFIDENCE AND SUPPORT FOR ALL DATA	45
FIGURE 4-8:ALL THE ACCEPTED RESULTS.....	45
FIGURE 4-9:CONFIDENCE AND SUPPORT FOR ACCEPTED DATA	46
FIGURE 4-10: RELATION BETWEEN SUPPORT, LIFT, TH=0.001	46
FIGURE 4-11:RELATION BETWEEN SUPPORT, LIFT, TH=0.1	47
FIGURE 4-12: THROUGHPUT WITH AND WITHOUT MAPREDUCE	48
FIGURE 4-13: ENERGY CONSUMPTION WITH AND WITHOUT MAPREDUCE	48

LIST OF TABLES

TABLE 1: OUR RESEARCH CRITERIA COMPARED TO PAIK ET AL., 2014.....	19
TABLE 2: FIRST RELATED STUDY CRITERIA.	26
TABLE 3: SECOND RELATED STUDY CRITERIA COMPARED TO SATOH, 2014.....	27
TABLE 4: THIRD RELATED STUDY CRITERIA COMPARED TO SATOH, 2016.....	27
TABLE 5: THIRD RELATED STUDY CRITERIA COMPARED TO (FARRAH, ZIYATI , & OUZZIF, 2015)	28

List of Abbreviations

DBaaS	Database as a Service
HDFS	Hadoop Distributed File System
IBM	International Business Machines
IoT	Internet of Things
KNN	k-Nearest Neighbors algorithm
MRIoT	MapReduce Internet of Things
NoSQL	Not Only Structured Query Language
PC	Personal Computer
RFID	Radio-frequency identification
SQL	Structured Query Language
WSN	Wireless sensor network
XML	eXtensible Markup Language

List of Published Papers

- Oweis, N. E., Owais, S. S., George, W., Suliman, M. G., & Snášel, V. (2015). A Survey on Big Data, Mining: (Tools, Techniques, Applications and Notable Uses). In *Intelligent Data Analysis and Applications* (pp. 109-119). Springer International Publishing. Indexed in Scopus.
- Oweis, N. E., Aracenay, C., George, W., Oweis, M., Soori, H., & Snasel, V. (2016). Internet of Things: Overview, Sources, Applications and Challenges. In *Proceedings of the Second International Afro-European Conference for Industrial Advancement AECIA 2015* (pp. 57-67). Springer International Publishing. Indexed in Scopus.

CURRENT PAPER:

- Big Data Analytics for the Internet of Things Based-on MapReduce.

Developing Parallel Technique for the Internet of Things Devices by using local MapReduce at the sink node.

By
Muna G. Oweis

Supervised by:

Dr. Sadeq AlHamouz

ABSTRACT

Due to the accelerating demand on the Internet of things (IoT) objects in which the Wireless Sensor Network (WSN) is the main source of Big Data, the huge scaled data is gathered excessively, causing network traffic problems and consuming huge amounts of power and an enormous size of memory which, in turn, would affect the network's performance. In order to address the issues of network performance related to Big Data, several research studies were conducted in an attempt to provide both convenient and effective solutions for such issues.

This thesis aimed at providing parallel, localized technique for IoT devices utilizing MapReduce at the sink node rather than employing this technique to agents' wireless or storage devices. In this study, the dataset was generated using a Java code, and MATLAB software programming tools (Math Works, R2015a) were used. MapReduce was used twice in order to manage big data; the first time for producing key value pairs, and the second time for reading pairs on the sensor to produce all distinct reading.

The results showed that the MapReduce approach utilized in this work resulted in less power consumption, less network traffic, and more efficient memory usage. MapReduce outperformed the traditional protocols by (Paik, Nam, Kim, & Won, 2014). The data reduction by utilizing the MapReduce approach was found to reach 79% in comparison to the 63% reported by others in the literature. We also found 43% enhancement of throughput and 27% less energy consumption with MapReduce compared to traditional protocols in WSNs.

Keywords: Internet of Things; Big Data; Map Reduce; Wireless Sensor Networks; Data Mining; Throughput; Energy consumption.

(تطوير تقنيات موازية لأجهزة أنترنت الأشياء باستخدام متعقب الوظيفة المحلي في عقدة التجميع)

إعداد

منى جورج عويس

إشراف

الدكتور صادق الحموز

ملخص

بناءً على الطلب المتسارع و المتزايد على شبكة أنترنت الأشياء (Internet of Things) حيث تعتبر شبكة الاستشعار اللاسلكية (WSN) هي المصدر الرئيسي للبيانات العملاقة (Big Data) حيث ان الزيادة المفرطة في تجميع البيانات ونتاجها قد تؤدي الى مشاكل في أداء الشبكة وخسارة الطاقة والذاكرة مما يؤثر على أداء الشبكة ، ولهذا الغرض عمل عدد من الباحثون لإيجاد حلول ملائمة وفعالة لحل العديد من تحديات البيانات العملاقة.

يهدف هذا البحث الى توفير تقنية موازية لأجهزة أنترنت الأشياء باستخدام متعقب الوظيفة (Map Reduce) المحلي في عقدة التجميع ، مما يؤدي الى استهلاك طاقه أقل وتخفيف الحركة المروية على الشبكة واستخدام الذاكرة بشكل أكثر كفاءة على خلاف التقنيات المطبقة على أجهزة الاستشعار وأجهزة التخزين الرئيسية. ففي هذه الدراسة تم انتاج مجموعه بيانات باستخدام لغة البرمجة جافا و استخدام ادوات لغة البرمجة MATLAB وقد تم استخدام الـ Map Reduce مرتين في الاستخدام الاول تم انتاج ازواج من القيم المفتاحية و الاستخدام الثاني لقراءة الازواج لكل مستشعر لإنتاج القراءات المتميزة .

حيث اظهرت نتائج هذه الدراسة مقارنة مع احدث الدراسات التقليدية الحالية (Paik, Nam, Kim, & Won, 2014) والتي استطاعت تقليل كمية البيانات المرسله تقريبا الى 63% في حين اثبتت هذه الدراسة على تقليل كمية البيانات المرسله تقريبا الى 79%، وأيضا على توفير الطاقة لتصل تقريبا الى 27% وتحسين الانتاجية لتصل الى 43% وذلك بدمج تقنيات متعقب الوظيفة مع تقنيات استخراج البيانات (Data Mining).

الكلمات المفتاحية: أنترنت الأشياء، البيانات العملاقة، متعقب الوظيفة، شبكة الاستشعار اللاسلكية، استخراج البيانات، الانتاجية، توفير الطاقة.

Chapter One

Introduction

1. CHAPTER ONE: INTRODUCTION

1.1.OVERVIEW

Nowadays, the Internet of Things (IoT) is growing fast and the fourth industrial generation (Industry 4.0) has just begun and became realistic. Billions of new physical devices, such as smart devices and Wireless Sensor Networks (WSN) are expected to be connected in the near future. WSNs are available in various applications such as healthcare, military within multiple organizations and institute. Therefore, the data gathered and collected from the WSN are considered to be a great source of Big Data. With the huge communication technology, more data are generated and collected, therefore, the Big Data will grow exponentially and this will make the challenges to extract and retrieve hidden valuable data more complex (Fuad, Oweis, Gaber, Ahmed & Snasel, 2015).

Currently, there are more than two billion users of smart objects including smartphones, smart homes, as well as business and entertainment applications (Miorandi, Sicari, De Pellegrini, & Chlamtac, 2012; Gartner's, 2014). These smart devices allow Machine to Machine (M2M) electronic communication with or without a user-intermediar. This has led to what is known as the "Internet of Things (IoT)" (Oweis.et al., 2016). In addition to that, the modern smart devices are creating a huge stream of structured, semi-structured and even unstructured data which eventually leads to increased data variety, increased storage capacity, and complex processing systems which is currently known as Big Data (Fouad, Oweis, Gaber, Ahmed & Snasel, 2015).

The huge amount of data generation has been helpful in various fields such as medical, social, commercial, industrial and scientific. However, this has been accompanied by several challenges related to the high volume of data storage, the complexity of stored data, and the

problem of hidden valuable data. Therefore, it became very important to utilize Big Data mining to be able to find meaningful relationships and be able to manage, extract and analyze the data (Larose, 2014)

1.2.PROBLEM STATEMENT

Data mining is one of the greatest tasks needed for the management of Big Data (Oweis, et al., 2016). The term “Big Data” includes a huge, complex, and abundant structured, semi-structure and unstructured, as well as, hidden valuable data that are generated and gathered from multiple smart resources such as IoT devices, WSNs and many more.

This thesis aims at developing a new parallel data mining technique for the IoT devices by using local MapReduce processing model at the sink nodes to filter, clean and discover the information from the data stream generated by WSN. This approach will help to reduce the huge amount of irrelevant data generated and minimize network traffic. The significance of this approach is that it provides high data compatibility between sink nodes and Big Data storage devices.

In this thesis, parallel big data mining techniques known as MapReduce will be used for data locality processing for IoT devices in the sink node before transforming data to be stored in Big Data storage such as clouds and datacenters. Therefore, this model will work as a distributed programming model for IoT objects based on MapReduce model to automatically process large datasets. Accompanying this approach comes a need to enhance different measurement tools like throughput and energy consumption compared to traditional protocols used in WSNs.

1.3.RESEARCH QUESTION

At the end of this research work, the following three main questions will be answered:

- How is the filtering process carried out to eliminate any existing noise from the huge amount of data generated from the IoT devices?
- How could we improve the performance of the data mining tools by using MapReduce for an association rule mining to be used as a local processing unit in the IoT devices to find the correlation data?
- How could we improve the performance of the data mining tools by using MapReduce in throughput and energy consumption?

1.4.RESEARCH OBJECTIVES

The main aims of this thesis project was to develop a distributed programming model for IoT devices based on MapReduce framework that will automatically process stream data generated from WSN, which is considered the main source of big data. This model will provide high compatibility between the IoT data devices and big data storage such as clouds computing in the parallel approach. Additionally, the following objectives can be achieved from the proposed algorithm:

- High-speed data stream processing over parallel execution.
- Minimal data motion (locally-based mining and processing at the physical sink node where the data stream is collected).
- Fault tolerance (automatic and easy data stream recovery based on MapReduce functionality).
- Filtering, cleaning, and reduction of the irrelevant data generated from the physical devices. Therefore, the data will be ready to transfer and store in Big Data storage

devices. Hence, the proposed model will provide a high compatibility between the sink nodes and the Big Data storage devices which help to solve one of the main Big Data challenges.

- Reducing the network traffic since the data will be processed, cleaned and ready enough to be transferred and stored in the Big Data storage such as clouds.
- High scalability over parallel execution analyzing by accessing the different new sources of IoT data.
- Energy consumption minimizing: Executing MapReduce is expected to reduce energy consumption associated with large clusters of data management in data centers, especially since it is working on the sink node location.

The proposed model is different from other previously studied MapReduce-based processing models in that it can locally process the IoT data at the sink node, rather than within high-performance server clusters and data centers. The proposed algorithm will measure the performance against a huge capacity of the IoT data stream to prove its performance.

1.5.RESEARCH MOTIVATION AND SIGNIFICANCE

Data mining have been widely applied in many aspects of our life, such as industry, social networks, health care, finance, communications and many more. Therefore, it is necessary to investigate its usability extensively in using it within the new emerging technologies such as; WSN, IoT, industry (4.0), and Big Data.

One of the main problems in the availability of Big Data is how to extract the valuable and meaningful knowledge from this huge and complex set of data collected from multiple sources such as WSN. This challenge has motivated us to explore a data mining algorithm dealing with extracting the valuable data from large stream data generated by the WSNs.

Most of the traditional data mining algorithms are not easy to handle this huge and complex amount of sensor data. In this research, a parallel-based MapReduce functions will be used as the main technology to provide a local mining technique for IoT device before transforming data to be stored in some Big Data storage such as cloud. By sending the useful, interesting, and clean data to the Big Data storage devices, the transmission bandwidth capacity will be reduced and a higher level of compatibility between IoT devices and Big Data storage is expected.

1.6.THESIS OUTLINE

The rest of this thesis is organized as follows:

Chapter Two: Covers the literature review and the previous work that considered WSN as a source of Big Data.

Chapter Three: Presents the methodology used including the proposed work, dataset, software used, and the parallel based MapReduce algorithm steps.

Chapter Four: Illustrates the implemented system, with the performance evaluation and the evolution measurements including research results and discussion.

Chapter Five: Provides concise summary of the work and future research directions, limitations, and directions for future research.

Chapter Two

Theoretical Framework and Literature Review

2. CHAPTER TWO: THEORETICAL FRAMEWORK AND LITERATURE REVIEW

2.1 INTRODUCTION

This chapter presents background information about the main topics covered in this thesis along with the concepts used. Section (2.1) defines the big data concepts, characteristics, sources, and tools including the MapReduce for a parallel big data mining framework used in this thesis. Section (2.2) covers the Internet of Things (IoT) definition, and sources from both the hardware and software, application that employ IoT, and some of the main IoT challenges namely data mining and security.

2.2 BIG DATA CONCEPT

Big Data is a massive amount of data that is so complex to be managed with traditional database techniques, such as; Structured Query Language (SQL) and relational management database. Consequently, few Big Data analytic techniques, such as MapReduce, are used to process and manage the IoT along with the new industrial generation denoted as Industry (4.0) (Chen & Zhang, 2014).

Big Data is a huge, complex, multiple types, and un-relational datasets that is very hard to manage by using traditional techniques (Gartner, 2012). The main characteristics of big data included the 3 Vs characteristics (Veracity, Viability, and Value) and then was elaborated to include the following characteristics known as the 6 Vs:

- Volume: Describes the huge data size.
- Velocity: Describes the data communication, processing speeds per time unit.

- Variety: Describes the different data types (structured, semi-structured, and unstructured).
- Veracity: Describes the data quality, such as data cleaning, filtering.
- Viability: Describes the prediction possibilities.
- Value: Describes the valuable data knowledge

Big Data can be categorized by its 6Vs characteristics as shown in Figure (2-1) (M. Schroeck, R. Shockley, Smart, Romero-Morales, Tufano, 2012 ; Elorieknilans, 2014 ; Chen & Zhang, 2014 ; Ding et al ., 2014).

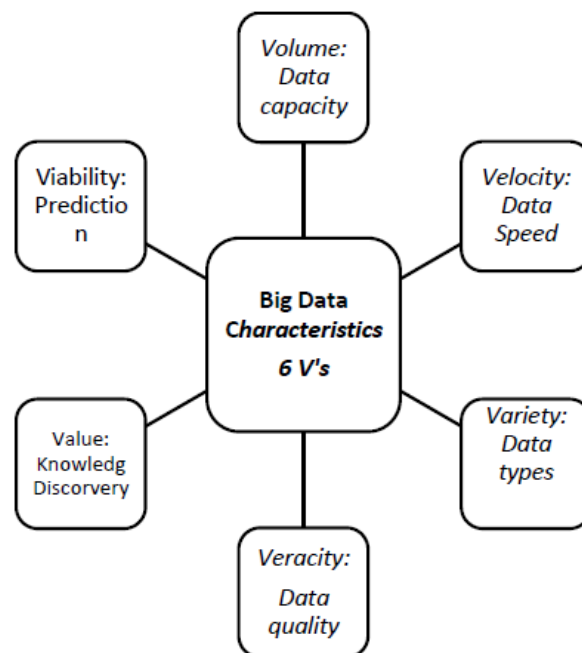


Figure 2-1: Big Data characteristics

2.3 BIG DATA TOOLS

In this section, the thesis mentions some of the currently tools used to manage the Big Data, also describes how these tools handle the complexity with all different types of data: structured, semi-structured, and unstructured. The tools mentioned in this section, can handle huge data volume, its handling capacity can vary between peta (10^{15}), to almost yotta (10^{24}) of bytes (Zakir, Seymour, & Berg, 2015). Google, Microsoft, and IBM are a good example for companies who are dealing with Big Data. They utilize these Big Data tools to extract, process, store, and analyze their data stream. This section also introduces some of the latest software and platforms developed to manage both the IoT and Big Data, such as: Not Only Structured Query Language (NoSQL), MapReduce and Hadoop as shown along with their types of utilization in Figure (2.2) (Wu, X, Zhu, X, Wu, G, & Ding, 2014 Barbierato, Gribaudo., & Iacono, 2014).

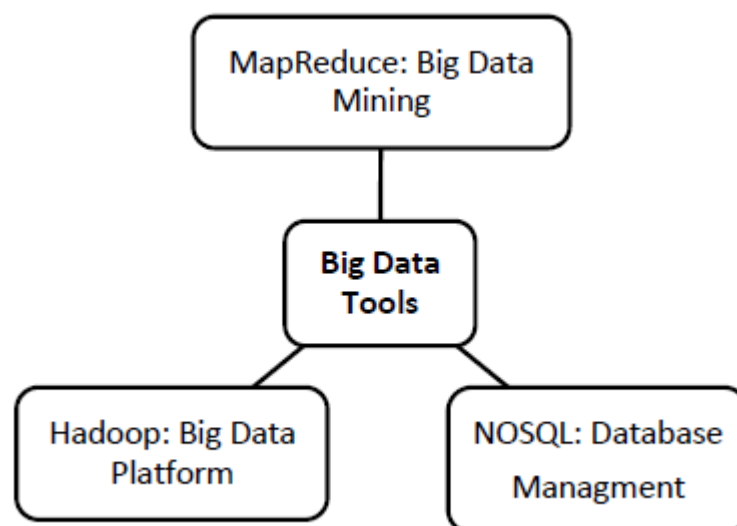


Figure 2-2: Big Data tools.

2.3.2 Not Only Structured Query Language (NoSQL)

NoSQL is an open source database software that is useful for Big Data management (Barbierato et al., 2014). NoSQL is combined with other tools like massive parallel processing, columnar-based databases and Database-as-a-Service (DBaaS), some examples of where NoSQL is used are the social media networks such Facebook, LinkedIn and Twitter which use Apache Cassandra NoSQL database tool (Lee, Park, & Lee, 2014)

NoSQL tool consists of two main parts: the traditional Structured Query Language (SQL) techniques, and the new queries to access and manage the large, complex, unstructured, and non-relational dataset that can be stored remotely on multiple virtual services in cloud dataset (Koch, 2013).

2.3.2 Hadoop

Hadoop is an open source software framework, freely available as a set of tools and libraries, based on Java to process, develop, and execute the large volume of distributed datasets and application (Zakir et al., 2015) , Hadoop splits the dataset into large chunks and distributes them among the nodes in a cluster. Hadoop can handle and execute thousands of terabytes of large, complex and non-relational dataset under several operating systems like Windows, Linux, BSD (UNIX), and OS X for Apple Macintosh (Domingo; 2012). Hadoop framework is also used by several online search engines such as Yahoo (Lee et al., 2014).

2.3.3 MapReduce

MapReduce is one of the Big Data mining techniques developed by Google that allows programmers to implement, process, and develop large dataset under parallel and distributed algorithms based on a local machine or clusters of machines (Satoh, 2014).

MapReduce can be implemented with different programming languages such as MATLAB, C, Java, and much more. (Lee et al., 2014; Zakir et al., 2015). It consists of two main functions: Map and Reduce functions. The Map is a function for dividing, filtering, and analyzing in the distributed cluster, while the Reduce summarizes the results into a single mode at a time demonstrated in Figure (2.3) (Srinivasa., & Bhatnagar, 2012).

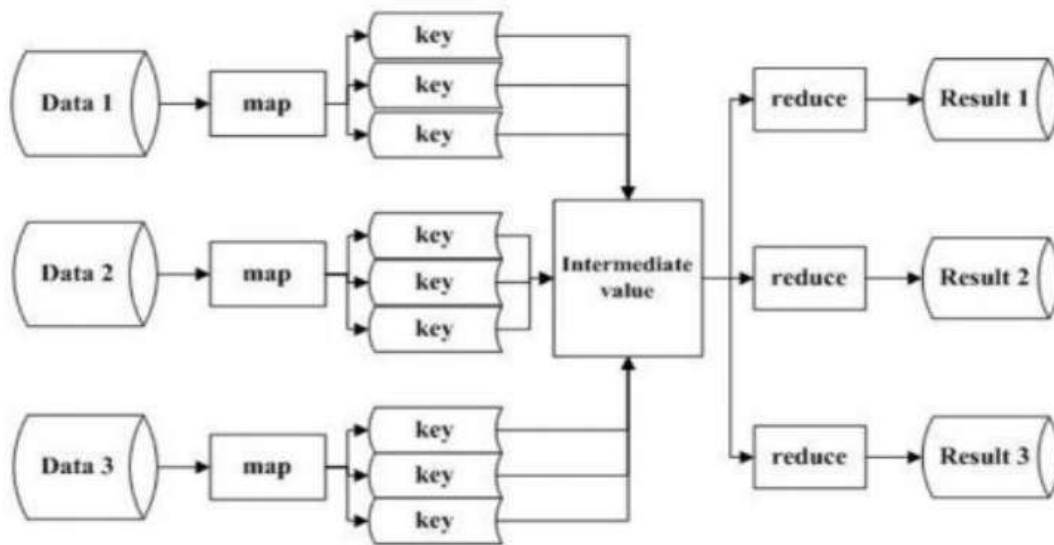


Figure 2-3: Basic Structure of the MapReduce process (Niu, Z., et al., 2012)

MapReduce can be applied to huge amounts of data to process either locally or in a cluster with a large number of servers. MapReduce can be used to sort a petabyte of data, within only a few hours.

Parallelism gives some possibilities of partial recovery from server failures if the operating portion, which produces a pre-processing operation or convolution fails (Koch, 2013), or its operation may be transferred to another working unit (assuming that the input data for the ongoing operation are available) (Wu et al., 2014). The most popular open-source implementation of the MapReduce is the Apache Hadoop software (Sangavi, Vanmathi, Gayathri, Raju, Paul & Dhavachelvan, 2015).

In this thesis, parallel-based MapReduce functions were used as the main engine that provided mining techniques executed locally for Big Data parallelism. In general, it can be applied to local or cluster base machines to handle the data stream generated from the IoT devices.

2.4 ASSOCIATION RULES

There are multiple data mining techniques that have been proposed for extracting valuable knowledge from a huge set of data. This is achieved by classification, clustering, association rules, prediction, sequential patterns and decision tree. Based on our interest in finding the relationship among a set of huge amount of sensors data, planned to use the association rule mining tool as one of the most significant data mining techniques to extract the knowledge from the huge amount of data collected in the sink node.

In association rule, a pattern is discovered based on a relationship between items in the same transaction. It is worth mentioning that the association techniques are widely used in several areas, such as market basket analysis, financial analysis, communication, health care and many other aspects (Hipp & Nakhaeizadeh, 2000).

Association rule mining is generated by two steps: Firstly, find all strong itemsets that have minimum support and/or minimum confidence (frequent itemsets, also called large itemsets). Secondly, use frequent itemsets to generate rules (Tan, Steinbach, & Kumar, 2005).

To extract the correlation between the itemset, the association rule mining algorithms focus on either sequential or centralized environment. For example, many of the IoT applications that utilize a large number of nodes. For example, WSNs have different distributed nodes among various locations, and each node has its own data. Merging datasets from these different sites into a centralized site can cause huge network communication costs and storage memory. Furthermore, finding frequent itemsets in parallel system is a significant problem;

because the number of individual transactions can be very large and usually will not fit in memory (shared memory). For this reason, this approach will be developing to handle a huge amount of data over parallel execution by using MapReduce as a big data analytics technique to generate rules from different datasets spread over different sites. Most researches focused on using Distributed Association Rule Mining (DARM) either for dealing with communication cost or privacy rather than the mining of the knowledge (Tassa, 2014; Hiwale and Ponde, 2015). They usually do a local frequent itemset for each site at the WSN itself, and then unite all local frequent into global frequent itemset by sending all of these frequent to the big data storage such as cloud for analysis use. However, the problem in such approaches is that the final result contains some items that are not frequent itemset with a large number of the un-relational dataset. This thesis will propose a new approach that ensures to mine the most relational items by mining frequent itemsets from the sink node as a general point, the rule is considered to be interesting if it satisfies a user minimum support and user minimum confidence at the same time as shown in Equation (1) (Makani, Z., Arora, S., & Kanikar, P., 2013).

$$Supp(X) = \frac{\text{Number of transaction that contain the itemset } (X)}{\text{The total number of transactions } (N)} \quad \text{Equation 1}$$

Where:

X: Is a selected itemset with a defined condition of interest.

N: is the total number of the transitions that defines the dataset.

The confidence of a rule is the ratio of the number of transactions that include all items in the Y, as well as the in the itemset X (the support) to the number of transactions that include all itemset in the X as shown in Equation (2) (Scheffer, 2001).

$$\text{Confidence } (X \rightarrow Y) = \frac{\text{Supp}(X \cup Y)}{\text{Supp}(X)}$$

Equation 2

Where:

X: Is a selected itemset with a defined condition of interest.

Y: Is a selected itemset with a another condition of interest.

Supp(X): Is the support of the transactions that include all itemsets in X.

Supp(XUY): Is the support of the transactions that include all items in the Y, as well as the in the itemset X

2.1. INTERNET OF THINGS (IoT)

The IoT refers to connected sets of physical devices, such as WSN, camera system, medical and industrial equipment and much more (Gubbi, Buyya, Marusic, & Palaniswami, 2013; and Satoh, 2016). All these physical devices are collecting and generating a huge amount of non-stop data stream. The capability to transfer data with a high-speed transmission is very important since it serves different fields, such as companies, industries, and academics for their decision making and contribution (Whitmore, Agarwal, & Da Xu, 2014).

Haller, Karnouskos & Schroth (2009) also defined the term “IoT” as: "A world where physical objects are seamlessly integrated into the information network and where the physical objects can become active participants in business process".

In the next section, will review the current hardware devices used in the IoT including the WSN as the main source of Big Data.

2.1.1. INTERNET OF THINGS HARDWARE

Recently, big data has grown up vastly, and their storage media such as data centers, clouds, and other Big Data storage facilities are commonly used in multiple IoT hardware (Sab, 2015). Our main focus in this study is on Wireless Sensor Network (WSN) as the main source of big

data. Such IoT devices include Radio Frequency Identification tags (RFID), smartphones, and much more.

2.1.2. WIRELESS SENSOR NETWORK (WSN)

For the purposes of this study, the WSN is considered to be the main source for the Internet of things hardware devices since it is one of the most effective sources of the Big Data.

A WSN consists of several autonomous sensor devices that are used to monitor physical and environmental conditions such as temperature, pressure, etc. Currently, there are hundreds or even thousands of sensors interacting together for collecting data streams that are widely used in financial analysis, online trading, medical testing, and so on (Wu et al., 2014 and Gubbi et al., 2013).

In WSNs, each sensor nodes generates, collects, and sends their data stream either to a central receiver (called a sink node) or to another sensor(s) in the same network. Therefore, each node plays the dual role of data originator and data router in a multi-hop sensor network (Deepti , 2012). The following Figure (2.4) illustrates the WSN architecture where sensor nodes send the data to their sink node. This study proposes using the sink nodes to apply a big data mining techniques.

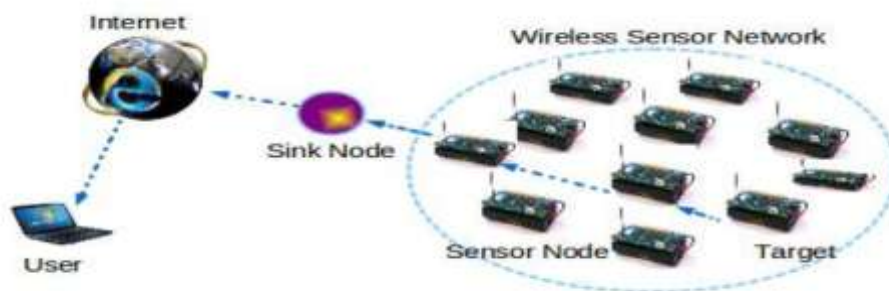


Figure 2-4: WSN Architecture (IIT Kharagpur., 2015)

WSNs can be characterized by unique features including: self-organizing capabilities, short range broadcast communication, dense deployment and cooperative effort of sensor nodes, and low-cost (Haenggi, Reuther, Goodman, Martinez, Ruiz, Nogueira & Al-Karaki , 2005).

In general, WSN design is refined by the need to maximize network lifetime and coverage while transmitting the chosen information with minimum energy consumption (Mahmood, Shi, Khatoon, & Xiao, 2013). Therefore, WSN was applied for monitoring in multiple applications such as health care, agricultural management, smart cities, and smart homes (Prasad, 2015) as shown in Figure (2.5).

As mentioned earlier, the WSN applications create a huge amount of a non-stop streaming data that faces several challenges to the WSNs in relation to storage limitation. The challenges include both power consumption and computational capabilities (Barbierato et al. 2014). The expected generated data from these WSNs applications make them the most important source of the Big Data.

In this thesis, a new parallel processing model was adapted for analyzing the data stream at the sink nodes of networks, before transforming it to the datacenter. Our proposed approaches is a unique approach in that there is need to process and analyze either in the node itself or in centralized storage devices such as cloud computing. However, the traditional techniques could not be directly applied to the WSNs applications. Hence, it is necessary to use a parallel technique such as " MapReduce" to handle some of the big data mining problems.



Figure 2-5: Illustration of the WSN application (libelium., 2015)

The WSN could be classified into two main categories: The centralized (sink node) and a distributed (sensor node) data processing. In this study work the processing was applied at the sink node.

XML is the standard data format for transmitting data through stream that are generated and collected by sensor nodes. Once the sensor nodes complete data collection round, it uploads the data as an XML file into the sink node, and therefore, the target data sets to be used in our mining are obtained from the sink nodes as shown in Figure (2.6).

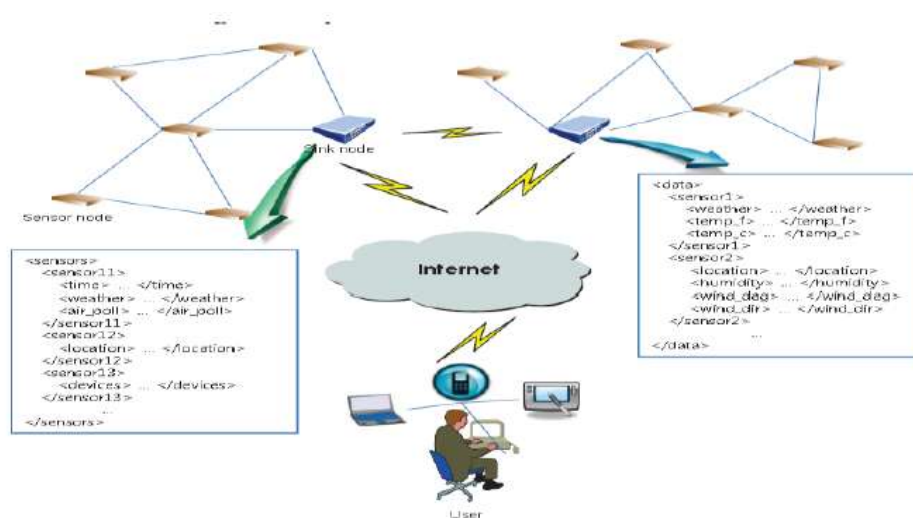


Figure 2-6: The WSN Structure (Paik et al 2014)

Finally, based on the presented literature, it can be concluded that the IoT is considered to be the most primary source of Big Data, including both the hardware and software applications. Therefore, this thesis purposes a parallel approach to obtain a high scalability over localized processing in the sink node to reduce the data size by transforming the correlation data only from this huge amount of data collected from the IoT devices.

2.5 LITERATURE REVIEW

Several studies have focused on MapReduce processing models for distributed programming model for IoT devices, such as; (Paik, Nam, Kim, & Won, 2014) who developed a comprehensive scheme for mining association rules from XML stream data. In this study work, the mining association rule scheme only requires one-time scan over parallel execution. The main intention of this proposed approach is to improve the performance of this study as shown in the Table (1) below.

Table 1: Our research criteria compared to Paik et al., 2014

Study Objective	Current Study Limitation	My proposed approach (MRIoT)
Filter, clean, and remove irrelevant data and extract the miningassociation rules for XML stream data.	Most of the currently proposed algorithms running either on the WSN side or in the cloud with traditional approaches which is not a suitable solution for big data. Hence, our new proposed algorithm run on the sink node.	The main intention of this proposal is to improve the performance of mining association node over parallel execution to avoid some of the association rule mining drawbacks. And by integrating the MapReduce approaches with association rule. This will provide a high scalability over parallel execution and fault tolerance

		(automatic and easy data stream recovery based on MapReduce functionality).
	Post processing is needed to limit the amount of huge data and get better mining results. This can be done by performing complex techniques such as pruning	Eliminate the post processing techniques with only one-time scan dataset.

2.6 RELATED WORK OVERVIEW

a) Studies concerned in MapReduce techniques within cloud computing

- ❖ Zhou & Huang (2014) improved the performance of Apriori algorithm based on MapReduce framework through parallel execution. The results analyzed the time of their enhanced parallel algorithm in correspondence with the already existing parallel algorithm. The proposed algorithm can handle some of the Big Data problems and achieves higher speed ratios. The association rules mining is one of the most significant features of data mining and it is expected to expose interesting relations between variables in the database. Yet a limitation of this study, the proposed algorithm (parallel Apriori algorithm) have just considered the check step which means, they only need to check the remaining $k - m$ subsets during candidate pruning.
- ❖ Ji et al. (2012) developed a parallel approach on real world applications based on MapReduce. The results cover the techniques used for Big Data in cloud computing, including the database and data storage. The study reviews the MapReduce that parallel handling system and applications, and finally present some of the current open issues and

challenges facing Big Data handling in cloud computing. As a conclusion of this study, most of the critical challenges of big data should be handled by developing multiple parallel algorithms which help in solving big data problems.

- ❖ Qian et al. (2015) proposed a parallel data reduction technique to be used as a pre-processing technique for big data by utilizing a MapReduce approach. The results showed that the reduction algorithms integrating with MapReduce that provide a high scalability in cloud computing by reducing the size which will help on both storage and network traffic. The study covered that most of the existing reduction algorithms currently cannot manage Big Data based on its highly costly so that the parallel approach is considered to be a suitable solution for data reduction in multiple fields as a pre-processing technique for big data.
- ❖ Triguero et al. (2015) proposed a novel system for reduction model techniques by using nearest neighbor techniques as a utilization approach for MapReduce. The results showed that the proposed system allowed reduction algorithms to be connected over Big Data. The results showed that this model was a suitable solution to improve the execution of the closest neighbor classifier with Big Data. As a conclusion of this study, by using the standard data mining tools in such huge datasets is not a suitable solution, therefore, it must be developed and improve to handle this new era of big data in cloud computing.
- ❖ Song, et al. (2015) presented the traditional algorithm: Genetic programming, which needs to be improved through parallel and distributed programming environment by utilizing the MapReduce approach. The experimental results showed that the enhanced parallel algorithm over traditional programming based on MapReduce has a preferable execution over the ordinary methodologies.

- ❖ Sangavi, et al. (2015) improved the execution performance of Dache (Distributed caching for .NET apps) by utilizing the MapReduce framework. The results showed that the unstructured information has been performed by utilizing the MapReduce approach in Hadoop framework, which provided an enhancement to the performance of Dache during the execution. This algorithm handled the unstructured data and eliminated the duplicate task which accompanies repetitive Mapping and Reducing tasks.

b) Studies concerned in processing big data with Hadoop in WSN

- ❖ Jung, Kim, Han, & Jeong (2014) proposed an architecture for HDSM (Hadoop-based appropriated sensor hub administration framework) for vastly distributed sensor hub administration by utilizing Hadoop through MapReduce approach and Distributed File System (DFS). It provided different techniques for gathering WSN data and transmitting them. In order to deal with numerous sensor hubs, certain MapReduce applications on sensor hubs were applied to facilitate the data transition from sensor hubs to DFS. Furthermore, it gave a flexible management scheme by re-configuring or updating configurations on the data formats of sensor hubs by using MapReduce. Finally, their test results demonstrated that their proposed method has more stability levels in term of performance.
- ❖ Jardak, Riihijärvi, Oldewurtel, & Mähönen(2010) presented the problems of storing and data collection from a huge amount of wireless sensor networks (WSNs). They showed the distributed database solutions such as BigTable and Hadoop are capable of dealing with storage of such huge amounts of data. They showed that MapReduce can indeed be used to develop such applications, and also describe in detail a general architecture for service platform for storing and processing data obtained from massive WSNs.

- ❖ Farrah, et al (2015) developed a data warehousing model for wireless network sensors to analyze all data gathered and detect the abnormal behavior by utilizing Hive Hadoop system based on MapReduce. The results showed that by utilizing the proposed system, Hive data warehouse software facilitated query and management of large datasets residing in distributed storage provides to Hadoop that provides the data collected by WSN. As a conclusion to this study is that reducing the size and low cost of the WSN has been extent the uses of these devices in the IoT area and most of the current database administration frameworks are not ready to manage and handle the WSN as the main source of Big Data. So that Big Data requires new improvements to process a huge volume of information generated by the WSN.
- ❖ Reyes-Ortiz, Oneto, & Anguita (2015) examined and compared two distributed cluster architectures: MPI/OpenMP, and Apache Spark on Hadoop. The latter provided a fast and general engine for big data processing with different cluster configurations. The results showed that MPI/OpenMP outperformed Spark by more than one order of amount in terms of processing speed and provides more consistent performance. However, Apache Spark shows better data management infrastructure and the possibility of working with other aspects such as node failure and data replication.
- ❖ del Río et al.(2014) demonstrated a comparison between a few strategies for excessive Big Data in Hadoop software. The results showed that serial algorithms are not a suitable way to deal with and manage Big Data and it is important to accommodate other parallel techniques such as the MapReduce approaches to handle big data challenges. And as a conclusion in this study, Big Data is considered now one of the most focused topics, but the current techniques of data mining for extracting suitable results methodologies are not ready to adapt to the new prerequisites forced by Big Data.

c) Studies concerned in Big Data within cloud computing

- ❖ Ding et al. (2014) presented a review of the current issues of Big Data in the wireless networking that covers the main characteristics of the big data including volume, velocity, variety, veracity, viability, and value (6 Vs) with its challenges as an open area of research along with the new emerging issues related to big data streams.
- ❖ Hashem et al.(2015) focused on the growth of the relationship between Big Data, cloud computing, and data storage frameworks in Hadoop platform. The results showed that multi-use, accessibility, quality, and security were administrative issues of the data that utilizes cloud administrations to store and handle the big data. They concluded that cloud computing is an effective innovation to performance enormous scale and complex processing of Big Data.
- ❖ Philip Chen and Zhang (2014) presented a survey on the main topics related to big data, including the applications, real-life opportunities and Big Data challenges in clouds such as data capture, data storage, data analysis and data visualization.

d) Studies concerned with Data mining techniques for WSN.

- ❖ Anadiotis, Morabito, & Palazzo (2015) proposed a system that gives a basic MapReduce approach for in the dynamic hub in the WSN. The execution of map and reduce functions is achieved by promoting the SDN-WISE protocol on both sides of the network: the controller and the sensor hubs. Their proposed method differentiates between different network topologies of processing large-scale and vastly distributed datasets.
- ❖ Bhavsar & Arolkar(2014) proposed a Multidimensional Association rule based on data mining technique for a cattle health monitoring system for extracting knowledge from WSNs as a main collector of the IoT data, they have additionally given a review of data mining principle with some choice of data mining technique utilized for WSNs data. At

the end, they proposed a rule based mining techniques for distinguishing different illness taking into account their side effects.

- ❖ Mahmood et al. (2013) covered how data mining algorithms are improved and enhanced to accomplish acceptable performance in WSNs. The proposed approach provided a data mining strategies that managed huge amounts of data generated by WSNs with its main challenges related to the association rule and finally conclude that most of the proposed techniques do not deal with the heterogeneous data and assume that the sensor data is homogenous.
- ❖ Lou, et al. (2014) Proposed a new algorithm denoted as Delivered Product Quality Rating (DPQR) to enhance the protection degree, time execution, and privacy by multi-parameters distributed databases for association rules.

e) Relevance Between Related Literature to Research Works

This section covers the mostly related previous work with respect to the proposed approach in this research:

- ❖ Paik, Nam, Kim, & Won (2014) generated frequent XML tree items without any redundancy. As a result, the size of the stream data was reduced to 37.5% of its original size. As mentioned before but with more details, Table (2) shows the main study objectives, and its limitations compared with our proposed approach.

Table 2: First related study criteria.

Study Objective	Study Limitation	My proposed approach (MRIoT)
Reformulation of association rules for XML streamed data for storing XML tree labels	Running with traditional approaches which are not suitable solution for big data	<p>Our proposed approach based on MapReduce and will achieve the following:</p> <ul style="list-style-type: none"> -Increased performance over parallel execution -Fault tolerance - Only one scan dataset -Reduced network traffic, because the data will be processed, cleaned and become ready in the sink node -Transfer of the clean, and related data only to be store in the Big Data storage such as clouds.

- ❖ Satoh (2014) Developed a new processing technique on the IoT clouds by using local MapReduce model at the edges nodes for data aggregation. The MapReduce model used in this study at the agent side as a distributed system for analyzing and processing large amount of parallel data. This system provided several advantages such as it reduced and removed the redundant data stored in IoT. Finally, the author used this system for real application for monitoring system to their office building to detect abnormalities from data measured by deferent matters, such as pressure and wattmeter. Table (3) shows the main study objectives, and its limitations compared with our proposed approach.

Table 3: Second related study criteria compared to Satoh, 2014

Study Objective	Study Limitation	My proposed approach (MRIoT)
Local MapReduce model at the edges nodes for data aggregation at agent side – distributed system of the cloud.	This system should be more compatible with Hadoop’s classes and interfaces, e.g., Mapper and Reducer, to directly reuse existing data processing software for Hadoop	Our proposed approaches will use the IoT devices through both wire and WSN.
Reduce the redundant data stored on the IoT devices	The current implementation assumes nodes are connected through wired networks	
No redundancy in generating frequent tree items		

- ❖ (Satoh, 2016) developed a distributed processing model based on MapReduce processing at the edge (sensor node) as a map operation, executed the program with their local data of network, IoT, and then gathered the results according to the user defining reduce operation at a sensor node. Table (4) showed the main study objectives, and its limitations compared with our proposed approach.

Table 4: Third related study criteria compared to Satoh, 2016.

Study Objective	Study Limitation	My proposed approach
Local MapReduce model at the edges nodes for data aggregation at agent side – distributed system of the IoT.	This system should be more compatible with Hadoop’s classes and interfaces, e.g., Mapper and Reducer, to directly reuse existing data processing software for Hadoop.	Our proposed approaches will use the IoT devices through a wired or wireless network.
Reduce the redundant data stored on the IoT devices	The current implementation assumes nodes are connected through wired networks.	
Noredundancyin generating frequent tree items.		

- ❖ Farrah, Ziyati , & Ouzzif (2015) developed a data warehouse model for WSN based on Hadoop platform by using Hive database strategies to analyze all data gathered, and to detect any abnormal behaviors. The author processed the data with Hadoop system. Table (5) shows the main study objectives, and its limitations compared with our proposed approach.

Table 5: Third related study criteria compared to (Farrah, Ziyati , & Ouzzif, 2015)

Study Objective	Study Limitation	My proposed approach (MRIoT)
Adopted the Hadoop framework to analyze and detect abnormal behavior from data gathered by WSN.	The current implementation focuses on deploying the solution on cloud environment and enhancing the real-time monitoring to detect anomalies	Our proposed approach will work locally on the sink node rather the cloud environment.
The modules used are HDFS as a storage platform, Hive as data warehouse and Oozie as workflow scheduler.		

SUMMARY

Several efforts were carried out to integrate methods for solving the Big Data problem from different perspectives and aspects while none of these efforts could get close to presenting a dynamic solution that can be considered as a guideline or complimentary work for our work. Hence, to the best of our knowledge of current research, and according to the excessive researches conducted for the relevant literature, our study work is unique with respect to the proposed technique and the outcomes are promising, moreover, considering the above literature, it is now obvious that MapReduce model based on parallel algorithms is one of the best choices as a high-performance IoT data mining technique. Therefore, this study proposes the application of the MapReduce model for a parallelized association rule algorithm for IoT

data. This model intends to process data at the sink node that stores the target data collected from sensor nodes, unlike other studies that either process data in Hadoop or in edge node (sensor node).

Chapter Three

The Methodology

3. CHAPTER THREE: METHODOLOGY

3.1.INTRODUCTION

The research methodology is based on studying and implementing the MapReduce approach, observing the performance of the algorithm with several parameters. This thesis aims to reduce the amount of data transmission to a big data storage, hence, reducing the transmission bandwidth, the amount of data in big data storage such as cloud computing, and improving the speed of transmission. Figure (3.1) shows the overall methodology design.

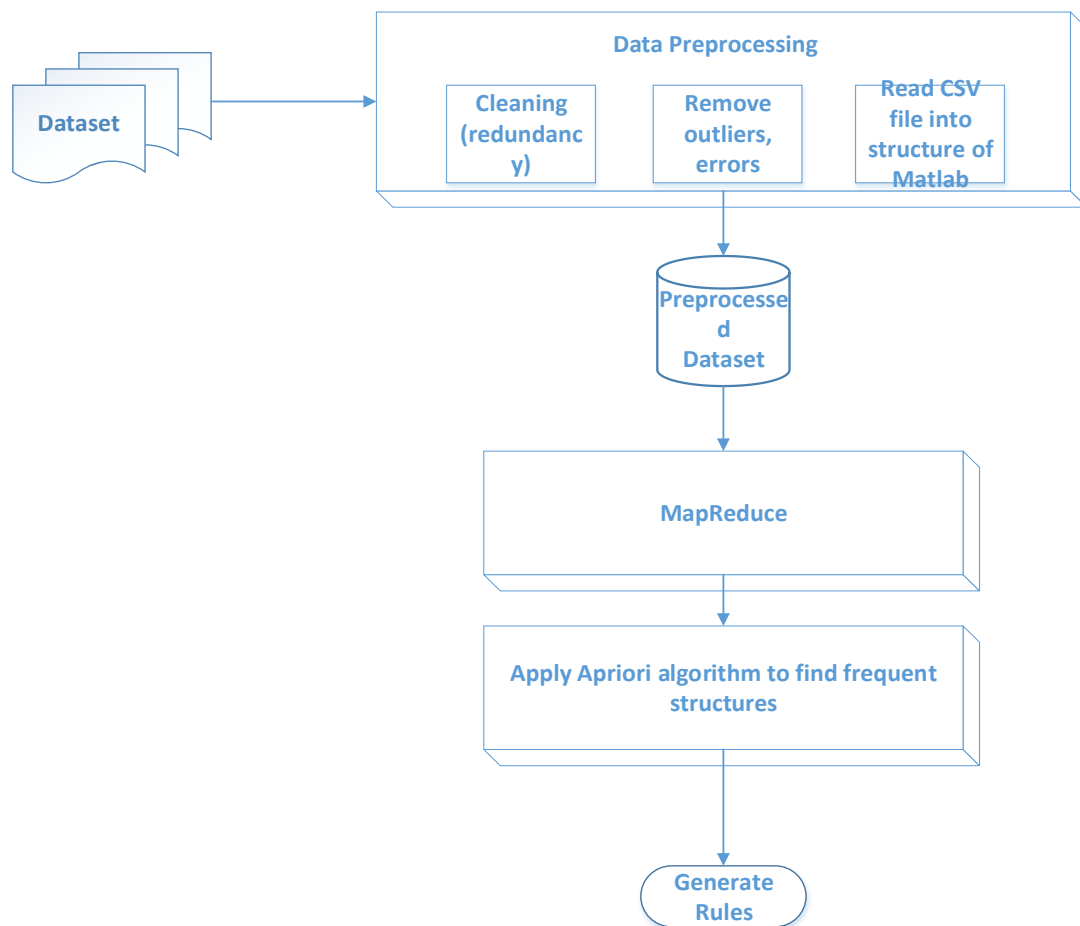


Figure 3-1: Methodology design for MRIoT.

To summarize, this study aims to apply an association rule mining techniques for the IoT devices by using local MapReduce processing model at the sink nodes to filter, clean and discover the knowledge of the data stream generated by WSN to help in reducing the irrelevant data generated, and network traffic as shown in the following general MapReduce system model (Figure (3.2)).

3.2.SYSTEM MODEL

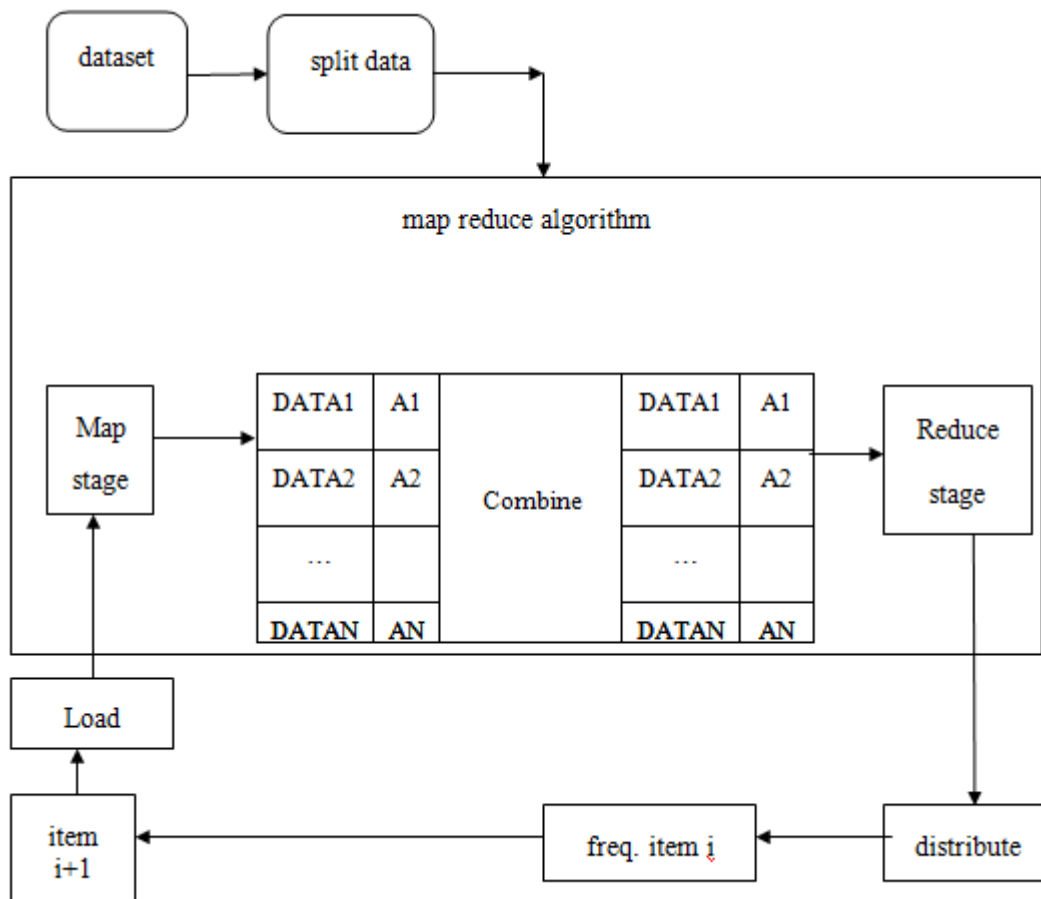


Figure 3-2: MapReduce System Model for MRIoT.

3.1 DATASET COLLECTION

To evaluate this study, developed a Java code to generate a dataset. The dataset contains a random reading value, for testing purposes.

3.2 SOFTWARE USED

In this study, MATLAB programming tools (Math Works, R2016a) was used as a high-level language for multiple reasons, such as:

1. MATLAB provides the ability to read and write the Comma separated files (CSV) data from the sink node because it has some built in functions for handling CSV file.
2. MATLAB provides an ability to convert CSV file into a MATLAB structure for easy access to the data.
3. MATLAB version (R2015a) supports Big Data processing, especially by using the MapReduce model, and use datastore function to access data that does not fit into memory.

3.3 APPLY MAPREDUCE FOR IoT (MRIoT)

This section describes the MapReduce model that was applied to IoT devices at the sink node, entitled “MapReduce for IoT” (MRIoT). MapReduce is a programming model designed to process and generate the big dataset. As mentioned before the main goal of this study work was to reduce the amount of data transmission generated from the WSN to be store at cloud or any other big data storage, Figure (3.3) shows the MapReduce MRIoT steps. As we mentioned in chapter 2 that MapReduce has three main functions (Dean, and Ghemawat, 2008):

- **Map:** at this step, the data are mapped into a paired structure (K, value), where K is the key value, and value is the input data associated with the key value. This step ensures that only one copy of the redundant value is processed.
- **Shuffle:** at this step, all the data associated with the same key are grouped to be processed at the same location.
- **Reduce:** reduce a set of intermediate values, which shares the same key into a smaller set.

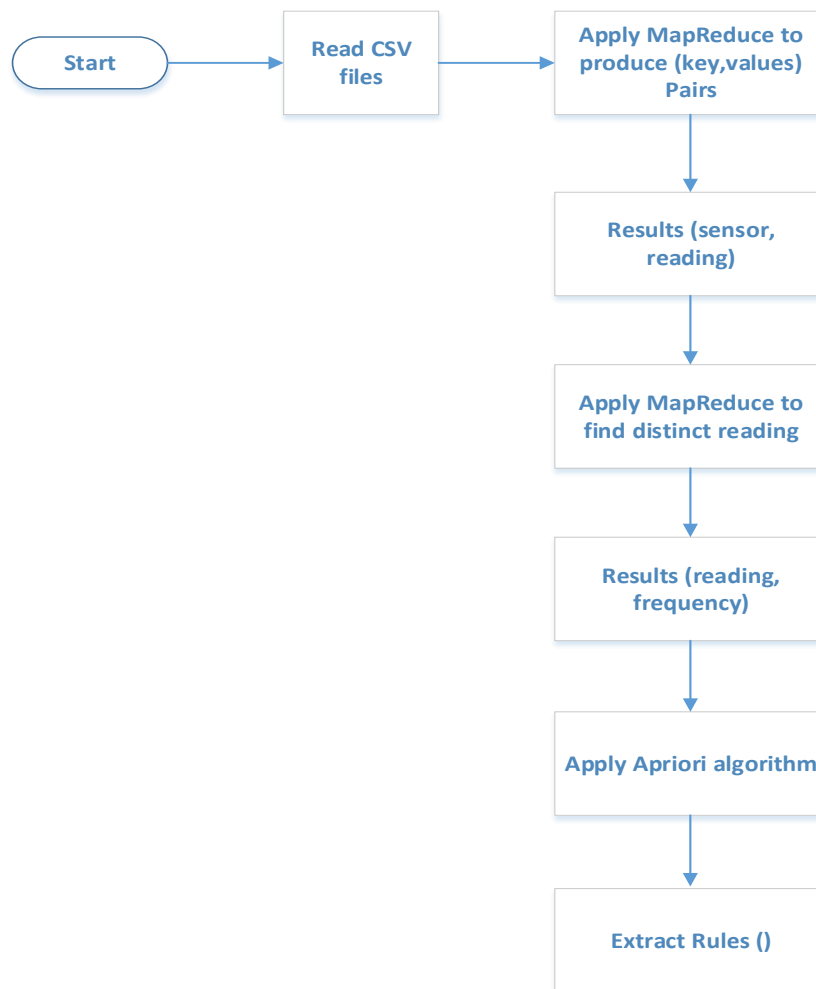


Figure 3-3: Flow work of the MRIoT.

In this methodology, we used MapReduce twice in order to manage big data; the first time we used it for producing key value pairs (sensor, reading), second time we used it again on the sensor, reading pairs to produce all distinct reading (reading, frequency). This accomplished by using MapReduce in two phases:

- At the first phase, map combined each reading with its corresponding sensor as pairs (k, v) , where k is the sensor id, and v is the reading value of the sensor, then at the reduce step, the reading values are combined for each sensor into a $(k, v[])$, where $v[]$ is a list of the reading values for the corresponding sensor, that ensures the uniqueness of the reading values.
- In the second phase, MapReduce was applied to the first phase results, to find all the distinct reading with their frequency.
- After the data has been preprocessed, we used association rule mining in order to identify the interesting relations between variables of large datasets.

The association rule mining aims to discover the knowledge of a large dataset, in our case, we will send the knowledge discovered by the association rule mining at the sink node instead of sending all the data streams of the sensor, hence reducing the amount of data transmission for a big data storage. For this purpose, we use Apriori algorithm, which is a breadth-first search algorithm used to identify the association rules that highlight the general trends of the database. Then apply the Apriori algorithm on the preprocessed dataset. The Apriori algorithm aims to identify the most frequent items, and prunes the infrequent items from the list, to generate the rules.

The following are the steps of the Apriori algorithm (Ye, and Chiang, 2006):

- 1- At the first step of this algorithm, each item is considered a member of the set of the candidate.
- 2- Generate second-itemset frequent pattern, the algorithm uses “Join” to discover the 2-itemset frequent pattern.
- 3- Generate third-itemset frequent pattern, this done by using Apriori property (all subsets of frequent itemset should be frequent), at this step a pruning step used to reduce heavy computation.
- 4- After determining all frequent subsets, the next step was to generate the association rules. The procedure for generating the association rules were as follow:
 - a) For each frequent itemset “ U ”, generate all nonempty subsets of U .
 - b) For every nonempty subset J of U , output the rule “ $J \rightarrow (U - J)$ ”

$$\text{If } support_count(U) / support_count(J) \geq min_conf$$

Where min_conf is a minimum confidence threshold, and the minimum support value of this study is 0.02, and the minimum confidence value is 0.01.

- 5- After the association rules generated by Apriori algorithm, only these rules sent to the data storage by the Internet instead of the huge data.

Pseudocode for Apriori algorithm (Wasilewska, 2007):

1. C_k : Candidate itemset of size k
2. L_k : frequent itemset of size k
3. $L1 = \{frequent\ items\}$;
4. **for**($k = 1$; $L_k \neq \emptyset$; $k++$) **do begin**
5. $C_{k+1} = candidates\ generated\ from\ L_k$;

6. **for each** transaction t in database **do**
7. increment the count of all candidates in $C_k + 1$ that are contained in t
8. $L_k + 1 = \text{candidates in } C_k + 1 \text{ with min_support}$
9. **end**
10. **return** $\cup_k L_k$

3.4 THROUGHPUT AND ENERGY CONSUMPTION

A lot of energy is consumed in communication of data and also in wireless sensor networks. The overall energy consumed contains the average dissipated energy through data transmission of all non-cluster nodes heads. Furthermore, the energy consumed through collection of data and summation of cluster head nodes is also important. Equations (3 and 4) below show how the procedure of L bits exchange in between the head nodes for the purpose of obtaining the amount of energy consumed.

$$E_{Tx}(d, x) = E_{elec} * L + \epsilon_{amp} * L \quad \text{Equation 3}$$

$$E_{Rx}(L) = E_{elec} * L \quad \text{Equation 4}$$

From the above equations, d is the sensor nodes distance, E_{Tx} is the value of consumed energy by the transmitter and E_{Rx} is the energy consumed at the receiving end. In addition, E_{elec} is the energy consumed by the electronics bit per bit both at the receiver and the transmitter nodes.

ϵ_{amp} is the energy consumed by the amplifiers at the sensor transmitting nodes and can be calculated using the formulae below:

$$\epsilon_{amp} = \epsilon_{fs} * d^2 \quad \text{For } d \leq d_0$$

$$\epsilon_{amp} = \epsilon_{mp} * d^4 \quad \text{For } d \geq d_0$$

ϵ_{fs} and ϵ_{mp} are the parameters of the energy communication.

- ✓ Energy usage: the aggregate amount of energy used by a sensor node in the whole process of communication. It is an indication of the rate at which energy source is consumed with time.
- ✓ Throughput-in any system, throughput is counted using Kbps and it is identified as a successfully packet received over a certain period.
- ✓ Each node has an implemented energy design that it uses for the purpose of obtaining the amount of energy consumed. Senility of events as well as data transmission is the major factors that result consumption of energy especially in wireless sensors.
- ✓ As a primary resource to many of the wireless sensor nodes, minimization of energy consumption is necessary. This can be done by ensuring that there are no unnecessary packets retransmissions. Energy Efficient Optimized Routing Algorithm has a better way of redundancy decrease of packets in several sections.
- ✓ The system performance is measured using throughput. It is indicated as aggregate of all the packets received over a certain period of time. Due to the fact that the amount of energy consumed by MapReduce with LEACH (Low-Energy Adaptive Clustering Hierarchy) is low, there is few dead nodes value within the system resulting to high throughput as compared to other designs. For the purpose of increased performance on the whole throughput networks, efficiency in data routing as well as a reliable path are offered by the MapReduce with LEACH. Reliability of sink path is as a result of increased throughput. The Equation (5) below shows how to find throughput.

$T = (\text{Successful Received Packets})/(\text{time})$	<i>Equation 5</i>
--	-------------------

Chapter Four

Implementation and Evaluation

4. CHAPTER FOUR: RESULTS AND DISCUSSION

4.1.INTRODUCTION

In this chapter, the performance of the MapReduce algorithm for wireless sensor network is evaluated. A dataset has been generated using java in order to apply testing. MATLAB (R2015 a) is used in this simulation since it contains a toolbox for MapReduce algorithm. Different practical scenarios can be achieved such as, the relation between confidence and support values in all transactions, accepted rules based on confidence value, throughput and power consumption with MapReduce algorithm.

4.2.ASSUMPTIONS

In the simulation, the following assumptions have been made:

- ✓ Each wireless sensor nodes are assumed to be homogeneous (i.e have same size and energy). The maximum energy of each node is assumed to be 1.8J.
- ✓ The nodes are random normally distributed within an area with $\mu = 0$ and $\sigma = 100$, where μ is the mean and σ is the standard deviation.
- ✓ The antenna type for sensor nodes is assumed to be omni-directional with antenna gain of 0 dBi.

4.3.SENSOR READING RESULTS

4.3.1. KEY AND VALUE

In IoT systems, counting is applied to locate the items within a set of data that have high frequency more than the one is allowed by the support. The easiest counting designs for implementation MapReduce structures are distributed and parallel counting since they involve division of data into chunks and later spread to various nodes with each node counted through parallel counting. The counting results are reduced in size and are then sent to the central node for overall sum calculation. Counting is referred as an addition operation since it involves different summations that compute the middle sums before they are sent to the network thus reducing the sizes transferred data through the network. The initial implementation criteria had two main challenges i.e. until all the repeated items are found, all the data sets had to be rescanned and also the hash tables that contain the references might be significant not fitting the memory allocated. Figures shown below are a representation of the process of implementing distributing counting through the use of Map Reduce structures. As shown below, key and value are used to merge or joining two or more datasets within a specific association rule. For example, in the following Figures (4.1, 4.2, 4.3, 4.4), key and values can be shown in different combinations, the key can be connected with 2 or more than value.

Key	Value
1: 100	
2: 1001	
3: 1002	
4: 1003	
5: 1005	
6: 1006	
7: 101	
8: 1011	

Figure 4-1: Key-Value results (key)

Key	Value
1: 100	1: 100
2: 1001	2: 1001
3: 1002	3: 1002
4: 1003	4: 1003
5: 1005	5: 1005
6: 1006	6: 1006
7: 101	7: 101
8: 1011	8: 1011

Figure 4-2: Key-Value results (value)

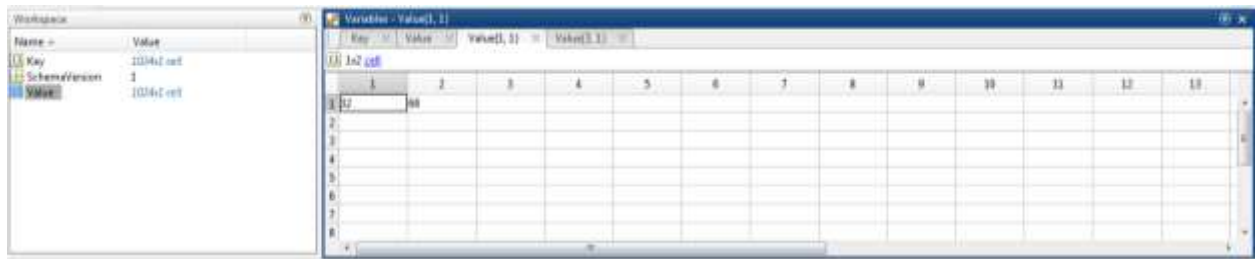


Figure 4-3: key for two values

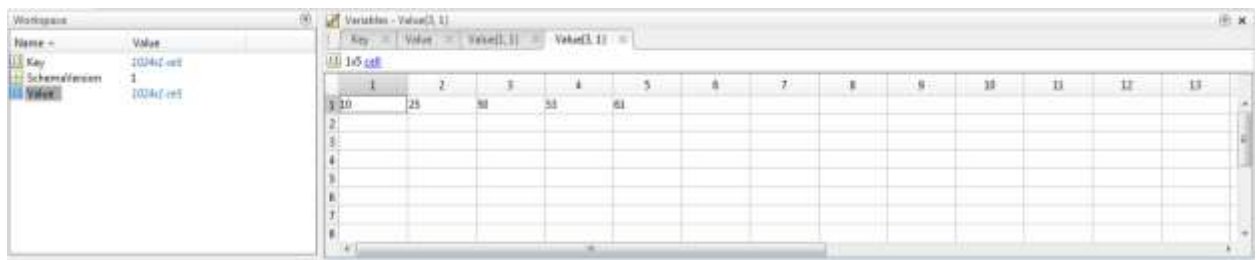


Figure 4-4: key for multi values

In order to observe the relation between key and value for all dataset, the following Figure (4.5) represents the relation between all keys and values.

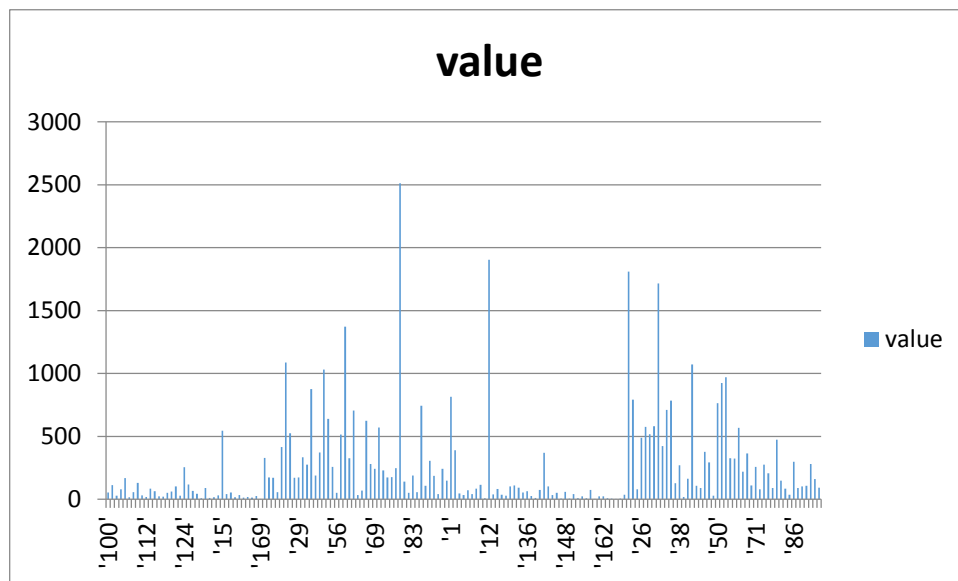


Figure 4-5: all keys and values

4.3.2. SYSTEM PARAMETERS

Association analysis is referred to as the Market base analysis, and also known as the item set mining frequency. Some of the basic concepts are support, lift, and confidence. The tiny data set is shown on the headers of the column demonstrates all the variables. An item set is referred to as any combinations of items in a subset. An item set contains items ranging from zero to all the other dataset details. In transactions, multiple item sets can be included. The number of appearances made by a particular item set in all the transactions is indicated in the support count. Support shows the frequency appearance.

$$\text{Support} = \text{item set support count} / \text{number of operations.}$$

Association rules contains of Confidence and lift.

Antecedents and consequents make the association rule and is represented as follows:

$$\{ante\} \Rightarrow conseq$$

Confidence is the core measure of rules, and that shows how many times particular standard can be used in a given transaction with the ante. Certain rules are applied when the items from all the ants and the counsels found in a particular transaction. This is similar to the inclusions of an item set. To calculate rule confidence support, metric item set is used. Therefore,

$$\text{confidence} = \text{itemset support} / \text{ante support}$$

Lift is another unit used to measure rules. It involves the comparison of ante probability and conseq occurrence together separately to the frequency observed in a given combination. There is no specific interaction between the ante and the conseq occurring when the value of lift is 1, and thus, the probabilities of the two occur separately. When the value of If is more than 1, there is an indication by the lift of the strength of both the ante and conseq dependability on each. This comparison can be used for respective support metrics.

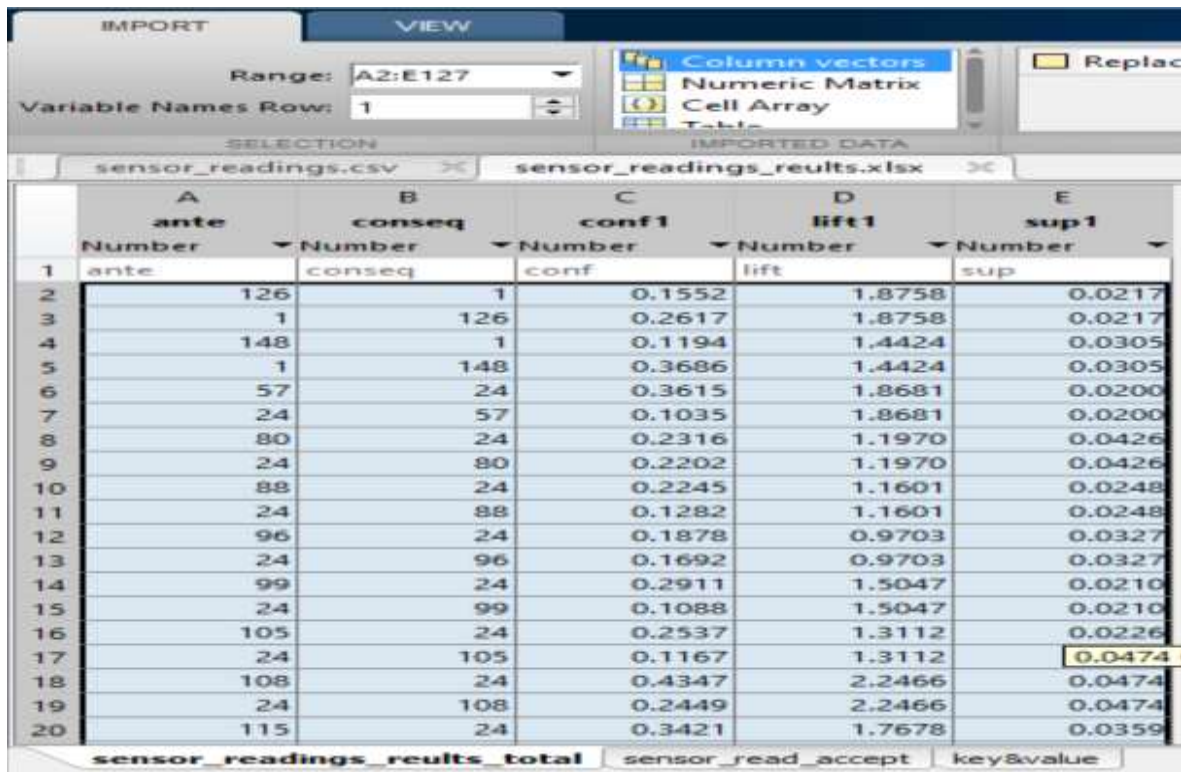
$$\text{Lift} = \text{itemset support} / (\text{ante support} \times \text{conseq support})$$

After recognising the fundamental ideas, it possible for us to identify the aim of our analysis: getting association rules with enough support level and confidence. To make the rules found as enjoyable as possible lift can be used as a secondary measure.

1. Creation of repetitive item set that can explain the support limit recurrently from item 1 to the maximum level removing the unnecessary candidates all the way.
2. Have a rule that explains the confidence limit in the same way

4.3.3. ALL RULES

To represent all the data items in a dataset, the MapReduce make use of the integer values for the purpose of fastening the design and also for efficient memory use. The process converting the data items into integer values involves delays and can be combined with a situation where repeated data items are of a certain size. Various entries are contained in a dataset, and each entry has a string of items. The addition of a unique integer value to each transaction is important for the purpose of obtaining a minimal initialization of data. The line numbers are taken as transaction id(TID), items are identified as integer representation (items ids) where the integer values replace the elements in both the column Ids and the row Id. The following Figure (4.6) represents all association rule with its confidence and support.



	A ante	B conseq	C conf1	D lift1	E sup1
	Number	Number	Number	Number	Number
1	ante	conseq	conf	lift	sup
2	126	1	0.1552	1.8758	0.0217
3	1	126	0.2617	1.8758	0.0217
4	148	1	0.1194	1.4424	0.0305
5	1	148	0.3686	1.4424	0.0305
6	57	24	0.3615	1.8681	0.0200
7	24	57	0.1035	1.8681	0.0200
8	80	24	0.2316	1.1970	0.0426
9	24	80	0.2202	1.1970	0.0426
10	88	24	0.2245	1.1601	0.0248
11	24	88	0.1282	1.1601	0.0248
12	96	24	0.1878	0.9703	0.0327
13	24	96	0.1692	0.9703	0.0327
14	99	24	0.2911	1.5047	0.0210
15	24	99	0.1088	1.5047	0.0210
16	105	24	0.2537	1.3112	0.0226
17	24	105	0.1167	1.3112	0.0474
18	108	24	0.4347	2.2466	0.0474
19	24	108	0.2449	2.2466	0.0474
20	115	24	0.3421	1.7678	0.0359

Figure 4-6: All data results

The relation between confidence and support can be shown below, for first 10 rules. As shown in Figure (4.7) below, the values of support are greater than values of confidence, also the value of support and confidence are matching on the same line.

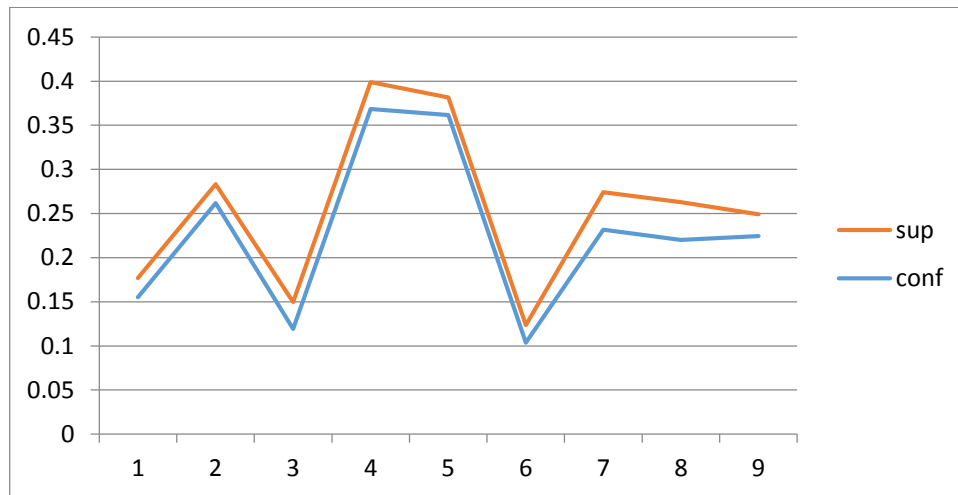


Figure 4-7: Confidence and support for all data

4.3.4. ACCEPTED RULES

The accepted results that are above specific threshold, can be shown in Figure (4.8) below, and the relation between support and confidence are shown in the next Figure (4.9).

IMPORT

VIEW

Range: A1:K20

Variable Names Row: 1

Column vectors

Numeric Matrix

Cell Array

Table

Replace

unimportable cells with

NaN

Import Selection

SELECTION

IMPORTED DATA

UNIMPORTABLE CELLS

sensor_readings.csv

sensor_readings_results.xlsx

	A	B	C	D	E	F	G	H	I	J	K
	VarName1	VarName2	VarName3	VarName4	VarName5	VarName6	VarName7	VarName8	VarName9	conf1	support1
	Cell	Cell	Cell	Cell	Cell	Cell	Cell	Number	Number	Number	Number
1										conf	support
2	{6} => {1}	—						6	1	0.1600	0.0200
3	{1} => {6}	—						1	6	0.2600	0.0200
4	{8} => {1}	—						8	1	0.1200	0.0300
5	{1} => {8}	—						1	8	0.3700	0.0300
6	{15} => {12...							15	12	0.3600	0.0200
7	{12} => {15...							12	15	0.1000	0.0200
8	{18} => {12...							18	12	0.2300	0.0400
9	{12} => {18...							12	18	0.2200	0.0400
10	{25} => {12...							25	12	0.2200	0.0200
11	{12} => {25...							12	25	0.1300	0.0200
12	{32} => {12...							32	12	0.1900	0.0300
13	{12} => {32...							12	32	0.1700	0.0300
14	{35} => {12...							35	12	0.2900	0.0200
15	{12} => {35...							12	35	0.1100	0.0200
16	{40} => {12...							40	12	0.2500	0.0200
17	{12} => {40...							12	40	0.1200	0.0200
18	{43} => {12...							43	12	0.4300	0.0500
19	{12} => {43...							12	43	0.2400	0.0500
20	{5} => {12}	—						5	12	0.3400	0.0400

sensor_readings_results_totol

sensor_read_accout

keyvalue

Figure 4-8: All the accepted results.

From workspace of MATLAB, number of accepted rules reached around 2056 while total number of data that worked under association rule around 9835. After applying MapReduce over the total accepted data under association rule and running the system 5 times and calculating the average, the reduction rate achieved was around 20.9%

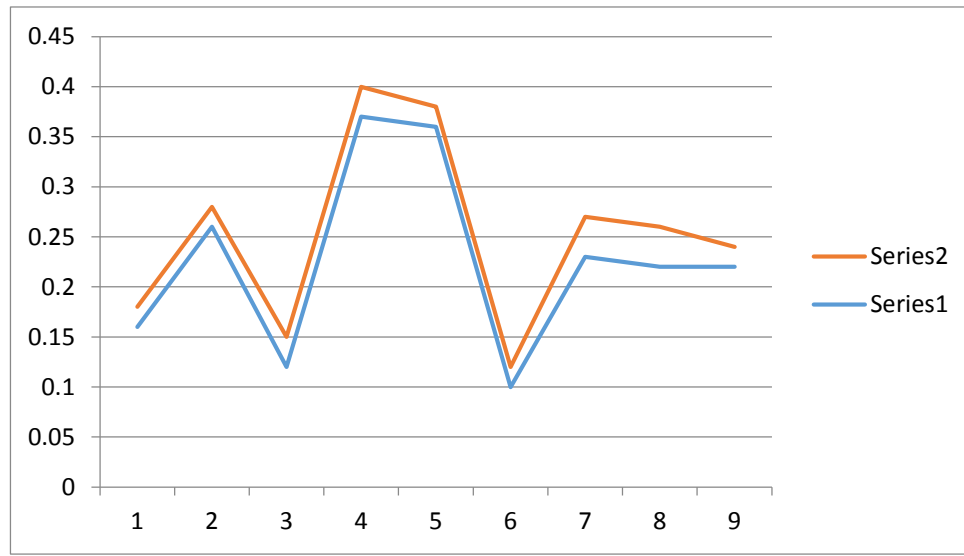


Figure 4-9: Confidence and support for accepted data

4.3.5. CASE STUDY 1 (TH = 0.001)

In the case of 0.001 threshold, the relation between support, lift and confidence are shown below in Figure (4.10), where most of the data concentrated in range of 0.2 and 0.4.

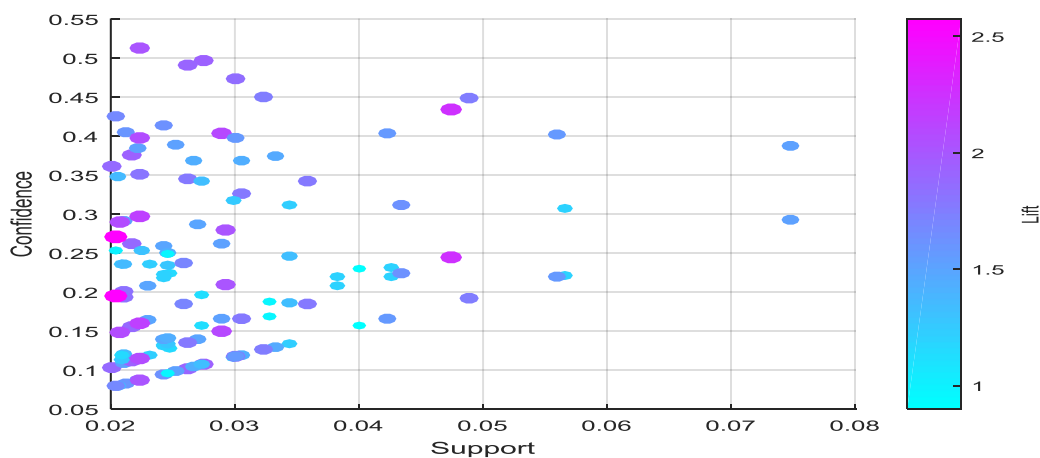


Figure 4-10: Relation between support, lift, TH=0.001

4.3.6. CASE STUDY 2 (TH = 0.1)

In case of 0.1 threshold, the relation between support, lift and confidence are shown below in Figure (4.11), where most of the data concentrated in range of 0.1 and 0.4, and there is concentration about threshold value.

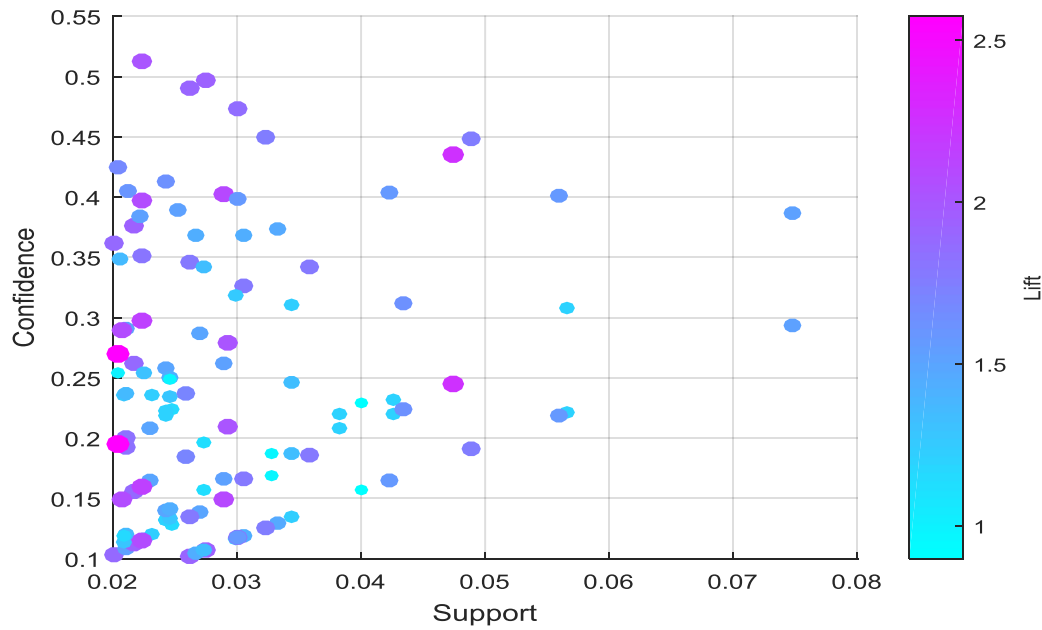


Figure 4-11: Relation between support, lift, TH=0.1

4.4. THROUGHPUT AND ENERGY CONSUMPTION RESULTS

By using LEACH protocol, the average throughput in Kbps are shown in Figure (4.12), in case of LEACH alone throughput (without MapReduce) reached around 25 Kbps, while after reduction data and enhance traffic at sink node, the throughput (with MapReduce) reached 44 kbps, in other words, the enhancement in throughput is shown below:

$$\text{enhancement} = \left| \frac{\text{with MapReduce} - \text{without MapReduce}}{\text{with MapReduce}} \right| = 43.18\%$$

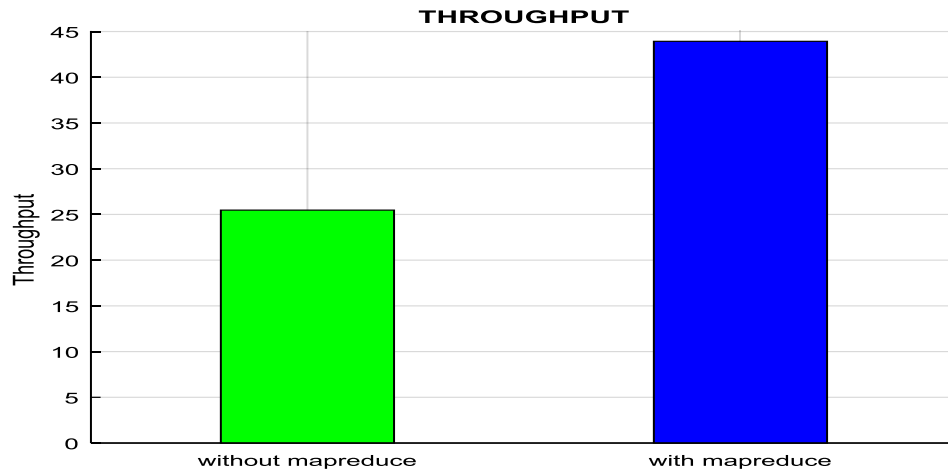


Figure 4-12: Throughput with and without MapReduce

In case of energy consumption, which assumed another criterion that has been selected to evaluate the performance of the proposed method. As shown in Figure (4.13), average energy consumption dropped from (3mJ) without MapReduce to (2.2mJ) with MapReduce, this can be used to save battery for WSN which can increase lifetime of overall system. the enhancement in energy consumption is shown below:

$$\text{enhancement} = \left| \frac{\text{without MapReduce} - \text{with MapReduce}}{\text{without MapReduce}} \right| = 26.66\%$$

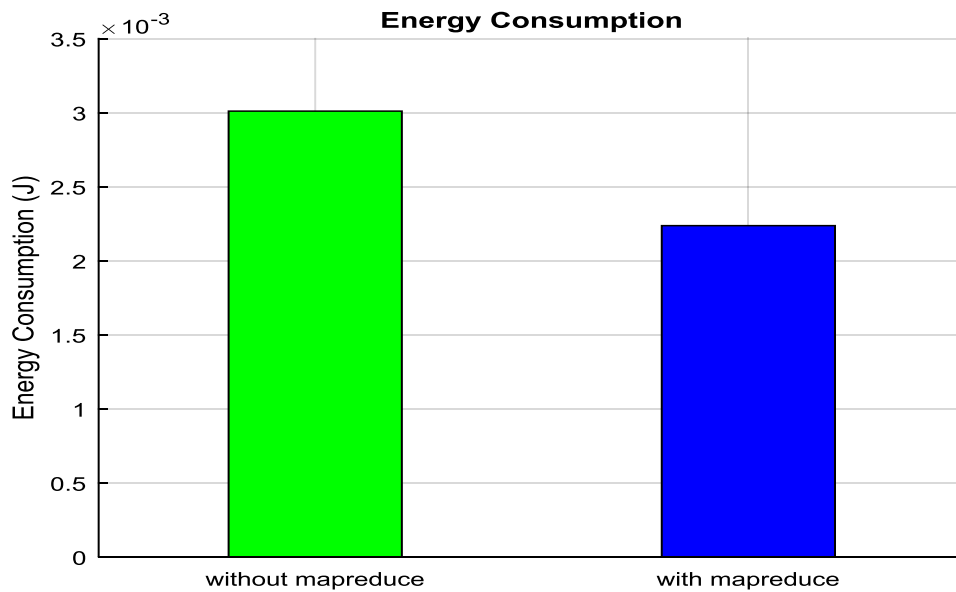


Figure 4-13: Energy consumption with and without MapReduce

The algorithm under which data is represented in the form of trees, and that has no requirement of tree balancing is known as XML data structure. This technique is a free form with some corelements such as the root containing a unique code that is distributed within the document, labeling of the internal nodes by use of tags while the leaves are identified through tags attributes. The ease of being intangible, as well as flexibility issues, makes XML be applied as the general format for data stream generation and transmission by the sensors in various WSN applications. However, different assumptions about stream data such the data streamsize taken as similar, and each block have an equal number of activities. Also, the sink nodes obtain useful data from the sensor nodes thus making the sensor nodes the primary target of our sets of data. Both the database entries and the construction units are identified as the major traditional association mining rules.

4.5. PERFORMANCE EVALUATION

Most of the existing research have investigated and introduced new methods in WSNs to enhance the performance of these networks. Due to the lower energy consumption in these networks, the large number of sensors that represented as nodes in these networks and the massive amount of time needed in the communication process between those nodes, new techniques must be developed in the way of covering all of these issues, especially in big data systems such as IoT.

In (Paik, et.al. (2014)) multiple developments have emerged in the field of WSN and that have resulted insignificant desperate volumes that are distributed depending on the geographic regions with heterogeneous data. The vast amount of data distributed has helped communities in their processes of data mining. Previous works of mining within the WSNs have concentrated much on the creation of relational data structures before the current works that require complex data structures. WSN's XML mining data have been facedwith several problems such continued flow of data as well as delicate tree structures.

The proposed algorithm covers the emerging definitions and strategies that are related to association rule mining that exceeds the XML data streams within. Mining stream data is different from the traditional mining data in several ways such as examination of each element of stream data at least once; this situation results in the application of online designs that needs full data scanning. The other difference is that the technique uses large memory sizes that are connected in spite of continued data regeneration. All the information elements within the stream data need fast processing while the results of generation of the online methods should be availed to all the users after each request is made. Various definitions exist, and that clearly define necessary information about association mining as well as the structure of XML data. Depending on the association rule being discussed, the rule is dependent on confidence and support as the primary measures. Support is used for the purpose of measuring of percentage in each transaction and contains X and Y elements while confidence is used to measure transaction rate within the Y items among all the transaction in D and that include X items.

To evaluate the performance of the proposed algorithm (MRIoT), data reduction rate has been compared to the one of the most recent research (Paik, Nam, Kim, & Won, 2014). The results show a significant difference in the reduction rate, with MRIoT being better. This confirms that the MRIoT algorithm can help in saving size of data compared to (Paik, Nam, Kim, & Won, 2014) the authors reached reduction from 100% to 37.5%, while in proposed system (MRIoT), it reached from 100% to 20.9%.

Chapter Five

Conclusions and Future Work

5. CHAPTER FIVE: CONCLUSIONS AND FUTURE WORKS

5.1 CONCLUSIONS

In this thesis, a parallel data mining technique for the IoT devices by using local Map Reduce was implemented in order to study the processing model at the sink nodes. The data was gathered via monitoring the data stream of different scenarios of WSN and measurements. The collected data maintained and processed locally at the Sink Node, handled and processed parallel to the relevant tasks. Another advantage was gained and utilized in this approach is the automatic fault tolerance by applying the Map Reduce.

This approach of parallel data mining using a local Map Reduce, minimizing the network traffic, which lead to speeding up network operation, and optimized the lookup and data retrieval process, this showed that it had reduced the huge amount of irrelevant data that would have needed unnecessary storage and handling. The significance of this approach is that it provides high data compatibility between sink nodes and Big Data storage devices without compromising the integrity of the data.

As shown from the results, the proposed algorithm in this thesis provides high efficiency at sink node by compressing and minimizing the total network data, and the results presented show the performance when using a local parallel Map Reduce in WSN it has a great benefit for both throughput and energy consumption.

Finally, different numerical results show the relation between total number of transactions in Map Reduce with accepted or rejected transactions.

5.2 FUTURE WORK

The future work includes the following:

- 1- Methods for reduction technique to be used as a preprocessing step to optimize the data process.
- 2- The evaluation considers only saving size of data with one destination (the sink node). Some improvements can be done with more general mechanisms such as with multiple sink nodes.

5.3 STUDY LIMITATION

The constraints of the MRIoT considered in this study work are the following:

- 1- The first is the real-time processing demand a very low-latency during the online processing at the sink nodes such as an online game.
- 2- The second is the data aggregation at the sink node. The collecting data from a multiple WSNs may not have all arrived at the same sink node which will reduce the efficiency of creating a high relevant correlation during the analytic process.
- 3- The third is the privacy. Accessing the sink nodes and perform an analytic process, especially in some critical areas such as military and health care mining will reduce the privacy record.

LIST OF REFERENCES

- Al-Karaki, M., and Kamal, A. Routing techniques in wireless sensor networks: A survey. *IEEE Wireless Communication*, Pp. 6-28, 2004.
- Anadiotis, A., Morabito, G., Palazzo, S., "An SDN-assisted Framework for Optimal Deployment of MapReduce Functions in WSNs", *IEEE Transactions on Mobile Computing*, no. 1, pp. 1, 2015.
- Bahsi, H. and Levi, A. (2010) 'Data collection framework for energy efficient privacy preservation in wireless sensor networks having many-to-many structures', *Sensors*, 10(9), pp. 8375–8397. doi: 10.3390/s100908375.
- Barbierato, E., Gribaudo, M. and Iacono, M. (2014) 'Performance evaluation of NoSQL big-data applications using multi-formalism models', *Future Generation Computer Systems*, 37, pp. 345–353. doi: 10.1016/j.future.2013.12.036.
- Bhavsar, A. R., & Arolkar, H. A. (2014). 'Multidimensional association rule based data mining technique for cattle health monitoring using wireless sensor network'. 2014 International Conference on Computing for Sustainable Global Development (INDIACom).
- Chong, S. K. Krishnaswamy, S. Loke, S. W. and Gaber, M. M. Using association rules for energy conservation in wireless sensor networks, in *Proceedings of the 23rd Annual ACM Symposium on Applied Computing (SAC '08)*, pp. 971–975, 2008.
- Dean, J., & Ghemawat, S. (2008). MapReduce: simplified data processing on large clusters. *Communications of the ACM*, 51(1), (pp.107-113).
- Del Río, S., López, V., Benítez, J.M. and Herrera, F. (2014) 'On the use of MapReduce for imbalanced big data using random forest', *Information Sciences*, 285, pp. 112–137. doi: 10.1016/j.ins.2014.03.043.
- Ding, G., Wu, Q., Wang, J., Yao, Y.-D. and Stevens (2014) 'Big spectrum data: The new resource for cognitive wireless networking'.
- Domingo, M.C. (2012) 'An overview of the Internet of things for people with disabilities', *Journal of Network and Computer Applications*, 35(2), pp. 584–596. doi: 10.1016/j.jnca.2011.10.015.

Domingue, J., Fensel, D. and Traverso, P. (2009) Future internet - FIS 2008 first future Internet symposium, FIS 2008, Vienna, Austria, September 29-30, 2008.

Farrah, S., El, H., El, M., Ziyati, H. and Ouzzif, M. (2015) 'An approach to analyze large scale wireless sensors network data', International Research Journal of Computer Science (IRJCS) Issue, 5(2).

Fouad, M. M., Oweis, N. E., Gaber, T., Ahmed, M., & Snasel, V. "Data Mining and Fusion Techniques for WSNs as a Source of the Big Data". Volume (65), (pp. 778-786), Procedia Computer Science, 2015.

Garg, D. and Garg, R. (2012) 'Angled-Leach in wireless sensor networks', International Journal of Advances in Computing and Information Technology, 1(2), pp. 144–152. doi: 10.6088/ijacit.12.10018.

Gartner, I. (2012) What is big data?. Available at: <http://www.gartner.com/it-glossary/big-data/> (Accessed: 26 February 2016).

Gartner's 2014 hype cycle for emerging technologies maps the journey to digital business (2014) Available at: <http://www.gartner.com/newsroom/id/2819918> (Accessed: 26 February 2016).

George, J. (2015) Hadoop MapReduce for mobile cloud. Available at: <https://books.google.jo/books?id=izCBrgEACAAJ> (Accessed: 26 February 2016).

Gottlob, G., Grasso, G., Olteanu, D. and Schallhart, C. (2013) Big data: 29th British national conference on databases, BNCOD 2013, Oxford, UK, July 8-10, 2013. Proceedings. Berlin: Springer.

Gubbi, J., Buyya, R., Marusic, S. and Palaniswami, M. (2013) 'Future generation computer systems Internet of things (IoT): A vision, architectural elements, and future directions', Future Generation Computer Systems, 29, pp. 1645–1660. doi: 10.1016/j.future.2013.01.010.

Haenggi, M., Reuther, A. I., Goodman, J. I., Martinez, D. R., Ruiz, L. B., Nogueira, J., ... & Al-Karaki, J. N. (2005). Handbook of sensor networks: Compact wireless and wired sensing systems. Opportunities and Challenges in Wire—less Sensor Networks. Boca Raton, FL: CRC Press.

- Gadeo-Martos, M.A., Fernandez-Prieto, J.A., Canada-Bago, J. and Velasco, J.R. (2011) 'An architecture for performance optimization in a collaborative knowledge-based approach for wireless sensor networks', *Sensors*, 11(12), pp. 9136–9159. doi: 10.3390/s111009136.
- Hashem, I.A.T., Yaqoob, I., Anuar, N.B., Mokhtar, S., Gani, A. and Ullah Khan, S. (2015) 'The rise of "big data" on cloud computing: Review and open research issues', *Information Systems*, 47, pp. 98–115. doi: 10.1016/j.is.2014.07.006.
- Hipp, J., & Nakhaeizadeh, G. (2000). Algorithms for association rule mining – A general survey and comparison. ACM58. Retrieved from http://kdd.org/exploration_files/hipp.pdf
- Jardak, C., Riihijärvi, J., Oldewurtel, F., and Mähönen P., Parallel processing of data from very large-scale wireless sensor networks. In *Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing (HPDC '10)*. ACM, New York, NY, USA, 787-794, 2010
- Ji, C., Li, Y., Qiu, W., Awada, U. and Li, K. (2012) 'Big data processing in cloud computing environments', 2012 12th International Symposium on Pervasive Systems, Algorithms and Networks, . doi: 10.1109/i-span.2012.9.
- Juliana, M.R. and Srinivasan, S. (2015) 'EA-LSDGS: Energy-aware location based secure data gathering scheme for wireless sensor networks', *Sensor Letters*, 13(12), pp. 1084–1090. doi: 10.1166/sl.2015.3599.
- Jung, I.-Y., Kim, K.-H., Han, B.-J., & Jeong, C.-S. (2014). Hadoop-Based distributed sensor Node management system. *International Journal of Distributed Sensor Networks*, 2014, 1–7. doi:10.1155/2014/601868
- Kaisler, S., Armour, F., Espinosa, J.A. and Money, W. (2013) 'Big data: Issues and challenges moving forward', 2013 46th Hawaii International Conference on System Sciences, . doi: 10.1109/hicss.2013.645.
- Knillans, E. (2014) The 5 vs of big data. Available at: <http://blogging.avnet.com/ts/advantage/2014/07/the-5-vs-of-big-data> (Accessed: 26 February 2016).
- Larose, D. T. "Discovering knowledge in data: an introduction to data mining". Second edition, John Wiley & Sons, 2014.

Lee, K.M., Park, S.-J. and Lee, J.-H. (eds.) (2014) Soft computing in big data processing. Springer International Publishing.

libelium smart word 50 sensor applications for a smarter world (no date) Available at: http://www.libelium.com/top_50_iot_sensor_applications_ranking/ (Accessed: 26 February 2016).

Li, Z. (2012) Computational intelligence and intelligent systems 6th international symposium, ISICA 2012, Wuhan, china, October 27-28, 2012. Proceedings. Berlin: Springer.

Loo, K. Tong, I. and Kao, B. Online algorithms for mining inter-stream associations from large sensor networks. in *Advances in Knowledge Discovery and Data Mining*, pp. 291–302, 2005

Lou, H., Yunlong, Zhang, F., Liu, M. and Shen, W. (2014) Data mining for privacy preserving association rules based on improved MASK algorithm. *Computer Supported Cooperative Work in Design (CSCWD)*, Proceedings of the 2014 IEEE 18th International Conference. Pp. 265 - 270

Ma, X. Li, S. and Luo, Q. Distributed, hierarchical clustering and summarization in sensor networks. in *Advances in Data and Web Management*, pp. 168–175, 2007

Mahmood, A., Shi, K., Khatoon, S. and Xiao, M. (2013) ‘Data mining techniques for wireless sensor networks: A survey’, *International Journal of Distributed Sensor Networks*, 2013, pp. 1–24. doi: 10.1155/2013/406316.

Makani, Z., Arora, S., & Kanikar, P. “A Parallel Approach to Combined Association Rule Mining”. Volume 62(15), (pp. 7-13), *International Journal of Computer Applications*, 2013.

MathWorksDocumentation, MapReduce

<http://www.mathworks.com/help/matlab/ref/mapreducer.html> (Last seen 12–Feb - 2016).

Miorandi, D., Sicari, S., De Pellegrini, F. and Chlamtac, I. (2012) ‘Internet of things: Vision, applications and research challenges’, 10(7), pp. 1497 – 1516.

Mishra, B.S.P., Dehuri, S., Kim, E. and Wang, G.-N. (2016) Techniques and environments for big data analysis. Available at: https://books.google.jo/books?id=6_WKCwAAQBAJ (Accessed: 26 February 2016).

Mohanty, S., Jagadeesh, M. and Srivatsa, H. (2013) Big data imperatives. Available at: <https://books.google.jo/books?id=WZdqU45XfkkC> (Accessed: 26 February 2016).

Novais, P., Camacho, D., Analide, C., Fallah Seghrouchni, A. El and Badica, C. (eds.) (2016) Intelligent distributed computing IX. Springer International Publishing.

Paik, J., Nam, J., Kim, U. and Won, D. (2014) ‘Association rule extraction from XML stream data for wireless sensor networks’, *Sensors*, 14(7), pp. 12937–12957. doi: 10.3390/s140712937.

Park, H., Shin, Y., Choi, S. and Kim, Y. (2013) ‘Symbolic and graphical representation scheme for sensors deployed in large-scale structures’, *Sensors*, 13(8), pp. 9774–9789. doi: 10.3390/s130809774.

Philip Chen, C.L. and Zhang, C.-Y. (2014) ‘Data-intensive applications, challenges, techniques and technologies: A survey on big data’, *Information Sciences*, 275, pp. 314–347. doi: 10.1016/j.ins.2014.01.015.

Philip Chen, C.L. and Zhang, C.-Y. (2014) ‘Data-intensive applications, challenges, techniques and technologies: A survey on big data’, *Information Sciences*, 275, pp. 314–347. doi: 10.1016/j.ins.2014.01.015.

Prasad, P. (2015) ‘Recent trend in wireless sensor network and its applications: A survey’, *Sensor Review*, 35(2), pp. 229–236. doi: 10.1108/sr-08-2014-683.

Qian, J., Lv, P., Yue, X., Liu, C. and Jing, Z. (2015) ‘Hierarchical attribute reduction algorithms for big data using MapReduce’, *Knowledge-Based Systems*, 73, pp. 18–31. doi: 10.1016/j.knosys.2014.09.001.

Rabl, T., Sachs, K., Poess, M., Baru, C. and Hans-Arno, J. (2015) Big data Benchmarking. Available at: <https://books.google.jo/books?id=FcHpCQAAQBAJ> (Accessed: 26 February 2016).

Report, E. (2012) IBM global business services business Analytics and optimization. Available at: https://www.ibm.com/smarterplanet/global/files/se__sv_se__intelligence__Analytics_-_The_real-world_use_of_big_data.pdf (Accessed: 26 February 2016).

S. Hiwale and S. Ponde, "Association rule mining in horizontally distributed database," *International Journal of Advanced Research in Computer Science*, vol. 6, no. 6, 2015.

Samarah, S. Al-Hajri, M. and Boukerche, A. A predictive energy-efficient technique to support object-tracking sensor networks, *IEEE Transactions on Vehicular Technology*, 60 (2), pp. 656–663, 2011

Sammer, E. (2012) Hadoop operations. Available at:
<https://books.google.jo/books?id=TQqSwRScVhoC> (Accessed: 26 February 2016).

Sandryhaila, A. and Moura, J.M.F. (2014) Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure. Available at:
<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6879640&isnumber=6879573>
(Accessed: 26 February 2016).

Sangavi, S., Vanmathi, A., Gayathri, R., Raju, R., Paul, P.V. and Dhavachelvan, P. (2015) 'An enhanced DACHE model for the MapReduce environment', *Procedia Computer Science*, 50, pp. 579–584. doi: 10.1016/j.procs.2015.04.087.

Satoh, I. (2014) 'MapReduce-Based data processing on IoT', (pp. 161–168). doi: 10.1109/iThings.2014.32.

Satoh, I. (2016). A Data Processing Framework for Distributed Embedded Systems. In *Intelligent Distributed Computing IX* (pp. 199-209). Springer International Publishing.

Scheffer, T. (2001). Finding association rules that trade support optimally against confidence. In *Principles of Data Mining and Knowledge Discovery* (pp. 424-435). Springer Berlin Heidelberg

Schmidt, K. and Phillips, C. (2013) Programming elastic MapReduce. Available at:
<https://books.google.jo/books?id=uBtRAgAAQBAJ> (Accessed: 26 February 2016).

Simulating a wireless sensor network (no date) Available at:
<http://vlssit.iitkgp.ernet.in/ant/ant/8/theory/> (Accessed: 26 February 2016).

Singh, A. and Rayapati, V. (2014) Learning big data with Amazon elastic MapReduce. Available at: <https://books.google.jo/books?id=-twkBQAAQBAJ> (Accessed: 26 February 2016).

- Singh, A. and Rayapati, V. (2014) Learning big data with Amazon elastic MapReduce. Available at: <https://books.google.jo/books?id=-twkBQAAQBAJ> (Accessed: 26 February 2016).
- Song, Z., Jun, M., Yang-Yang, Z. and Qiong, L. (2015) An improved parallel algorithm of genetic programming based on the framework of MapReduce. Available at: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7307816&isnumber=7307766> (Accessed: 26 February 2016).
- Srinivasa, S. and Bhatnagar, V. (eds.) (2012) Big data Analytics. Springer Berlin Heidelberg.
- Suthaharan, S. (2015) Machine learning models and Algorithms for big data classification. Available at: <https://books.google.jo/books?id=ad3HCgAAQBAJ> (Accessed: 26 February 2016).
- T. Tassa, “Secure mining of association rules in horizontally distributed databases,” Knowledge and Data Engineering, IEEE Transactions on, vol. 26, no. 4, pp. 970–983, 2014.
- Tan, P. N., Steinbach, M., & Kumar, V. (2005). Association analysis: Basic concepts and algorithms. In Introduction to Data Mining
- Triguero, I., Peralta, D., Bacardit, J., García, S. and Herrera, F. (2015) ‘MRPR: A MapReduce solution for prototype reduction in big data classification’, Neurocomputing, 150, pp. 331–345. doi: 10.1016/j.neucom.2014.04.078.
- Tseng V. S. and Lin, K. W. Energy efficient strategies for object tracking in sensor networks: a data mining approach, Journal of Systems and Software, 80(10), pp. 1678–1698, 2007
- Wasilewska, A. (2007). Apriori Algorithm. Lecture Notes, http://www.cs.sunysb.edu/~cse634/lecture_notes/07apriori.pdf, accessed, 10.
- Whitmore, A., Agarwal, A. and Da Xu, L. (2014) ‘The Internet of Things—A survey of topics and trends’, Information Systems Frontiers, 17(2), pp. 261–274. doi: 10.1007/s10796-014-9489-2.
- Wu, X., Zhu, X., Wu, G.-Q. and Ding, W. (2014) ‘Data Mining with Big Data’, 26(1), pp. 97 – 107.

Yang, Y., Duan, X. and Li, L. (2011) ‘Wireless sensor network time Synchronization design for large generator on-line monitoring’, *Sensor Letters*, 9(4), pp. 1467–1471. doi: 10.1166/sl.2011.1670.

Ye, Y., & Chiang, C. C. (2006, August). A parallel apriori algorithm for frequent itemsets mining. In *Software Engineering Research, Management and Applications*, 2006. Fourth International Conference on (pp. 87-94). IEEE.

Zakir, J., Seymour, T. and Berg, K. (2015) ‘BIG DATA ANALYTICS.’, 16(2).

Zhou, X. and Huang, Y. (2014) An improved parallel association rules algorithm based on MapReduce framework for big data. Available at:

<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6980847&isnumber=6980796>

(Accessed: 26 February 2016).

APPENDIX : RULES AND MAPREDUCE WORK

Key Value

—— ———

'100' [54]

'101' [112]

'102' [29]

'105' [79]

'106' [168]

minimum support = 0.02

Frequent Itemsets: 122

Max Level Reached : 3-itemsets

Number of Support Data : 2056

Minimum Confidence : 0.00

Number of Rules : 126

{6} => {1} Confidence: 0.16 Lift: 1.88 Support: 0.02
 {1} => {6} Confidence: 0.26 Lift: 1.88 Support: 0.02
 {8} => {1} Confidence: 0.12 Lift: 1.44 Support: 0.03
 {1} => {8} Confidence: 0.37 Lift: 1.44 Support: 0.03
 {15} => {12} Confidence: 0.36 Lift: 1.87 Support: 0.02
 {12} => {15} Confidence: 0.10 Lift: 1.87 Support: 0.02
 {18} => {12} Confidence: 0.23 Lift: 1.20 Support: 0.04
 {12} => {18} Confidence: 0.22 Lift: 1.20 Support: 0.04
 {25} => {12} Confidence: 0.22 Lift: 1.16 Support: 0.02
 {12} => {25} Confidence: 0.13 Lift: 1.16 Support: 0.02
 {32} => {12} Confidence: 0.19 Lift: 0.97 Support: 0.03
 {12} => {32} Confidence: 0.17 Lift: 0.97 Support: 0.03
 {35} => {12} Confidence: 0.29 Lift: 1.50 Support: 0.02
 {12} => {35} Confidence: 0.11 Lift: 1.50 Support: 0.02
 {40} => {12} Confidence: 0.25 Lift: 1.31 Support: 0.02
 {12} => {40} Confidence: 0.12 Lift: 1.31 Support: 0.02
 {43} => {12} Confidence: 0.43 Lift: 2.25 Support: 0.05
 {12} => {43} Confidence: 0.24 Lift: 2.25 Support: 0.05
 {5} => {12} Confidence: 0.34 Lift: 1.77 Support: 0.04
 {12} => {5} Confidence: 0.19 Lift: 1.77 Support: 0.04

{51} => {12} Confidence: 0.29 Lift: 1.48 Support: 0.03
 {12} => {51} Confidence: 0.14 Lift: 1.48 Support: 0.03
 {53} => {12} Confidence: 0.24 Lift: 1.22 Support: 0.02
 {12} => {53} Confidence: 0.12 Lift: 1.22 Support: 0.02
 {59} => {12} Confidence: 0.38 Lift: 1.94 Support: 0.02
 {12} => {59} Confidence: 0.11 Lift: 1.94 Support: 0.02
 {6} => {12} Confidence: 0.31 Lift: 1.61 Support: 0.04
 {12} => {6} Confidence: 0.22 Lift: 1.61 Support: 0.04
 {61} => {12} Confidence: 0.40 Lift: 2.08 Support: 0.03
 {12} => {61} Confidence: 0.15 Lift: 2.08 Support: 0.03
 {66} => {12} Confidence: 0.35 Lift: 1.81 Support: 0.02
 {12} => {66} Confidence: 0.12 Lift: 1.81 Support: 0.02
 {8} => {12} Confidence: 0.29 Lift: 1.51 Support: 0.07
 {12} => {8} Confidence: 0.39 Lift: 1.51 Support: 0.07
 {9} => {12} Confidence: 0.35 Lift: 1.79 Support: 0.03
 {12} => {9} Confidence: 0.14 Lift: 1.79 Support: 0.03
 {8} => {15} Confidence: 0.11 Lift: 1.95 Support: 0.03
 {15} => {8} Confidence: 0.50 Lift: 1.95 Support: 0.03
 {25} => {18} Confidence: 0.22 Lift: 1.19 Support: 0.02
 {18} => {25} Confidence: 0.13 Lift: 1.19 Support: 0.02
 {32} => {18} Confidence: 0.22 Lift: 1.20 Support: 0.04
 {18} => {32} Confidence: 0.21 Lift: 1.20 Support: 0.04
 {40} => {18} Confidence: 0.24 Lift: 1.28 Support: 0.02
 {18} => {40} Confidence: 0.11 Lift: 1.28 Support: 0.02
 {43} => {18} Confidence: 0.22 Lift: 1.21 Support: 0.02
 {18} => {43} Confidence: 0.13 Lift: 1.21 Support: 0.02
 {5} => {18} Confidence: 0.23 Lift: 1.27 Support: 0.02
 {18} => {5} Confidence: 0.13 Lift: 1.27 Support: 0.02
 {51} => {18} Confidence: 0.33 Lift: 1.77 Support: 0.03
 {18} => {51} Confidence: 0.17 Lift: 1.77 Support: 0.03
 {6} => {18} Confidence: 0.25 Lift: 1.34 Support: 0.03
 {18} => {6} Confidence: 0.19 Lift: 1.34 Support: 0.03
 {8} => {18} Confidence: 0.22 Lift: 1.21 Support: 0.06
 {18} => {8} Confidence: 0.31 Lift: 1.21 Support: 0.06
 {8} => {20} Confidence: 0.08 Lift: 0.99 Support: 0.02
 {20} => {8} Confidence: 0.25 Lift: 0.99 Support: 0.02
 {32} => {25} Confidence: 0.17 Lift: 1.50 Support: 0.03
 {25} => {32} Confidence: 0.26 Lift: 1.50 Support: 0.03
 {6} => {25} Confidence: 0.16 Lift: 1.49 Support: 0.02

{25} => {6} Confidence: 0.21 Lift: 1.49 Support: 0.02
 {8} => {25} Confidence: 0.13 Lift: 1.22 Support: 0.03
 {25} => {8} Confidence: 0.31 Lift: 1.22 Support: 0.03
 {8} => {27} Confidence: 0.10 Lift: 1.92 Support: 0.03
 {27} => {8} Confidence: 0.49 Lift: 1.92 Support: 0.03
 {8} => {3} Confidence: 0.09 Lift: 1.62 Support: 0.02
 {3} => {8} Confidence: 0.41 Lift: 1.62 Support: 0.02
 {8} => {30} Confidence: 0.08 Lift: 1.59 Support: 0.02
 {30} => {8} Confidence: 0.41 Lift: 1.59 Support: 0.02
 {8} => {31} Confidence: 0.08 Lift: 1.36 Support: 0.02
 {31} => {8} Confidence: 0.35 Lift: 1.36 Support: 0.02
 {40} => {32} Confidence: 0.24 Lift: 1.36 Support: 0.02
 {32} => {40} Confidence: 0.12 Lift: 1.36 Support: 0.02
 {5} => {32} Confidence: 0.20 Lift: 1.14 Support: 0.02
 {32} => {5} Confidence: 0.12 Lift: 1.14 Support: 0.02
 {51} => {32} Confidence: 0.26 Lift: 1.48 Support: 0.02
 {32} => {51} Confidence: 0.14 Lift: 1.48 Support: 0.02
 {53} => {32} Confidence: 0.25 Lift: 1.43 Support: 0.02
 {32} => {53} Confidence: 0.14 Lift: 1.43 Support: 0.02
 {6} => {32} Confidence: 0.20 Lift: 1.12 Support: 0.03
 {32} => {6} Confidence: 0.16 Lift: 1.12 Support: 0.03
 {8} => {32} Confidence: 0.16 Lift: 0.90 Support: 0.04
 {32} => {8} Confidence: 0.23 Lift: 0.90 Support: 0.04
 {8} => {35} Confidence: 0.10 Lift: 1.44 Support: 0.03
 {35} => {8} Confidence: 0.37 Lift: 1.44 Support: 0.03
 {8} => {36} Confidence: 0.11 Lift: 1.34 Support: 0.03
 {36} => {8} Confidence: 0.34 Lift: 1.34 Support: 0.03
 {8} => {40} Confidence: 0.13 Lift: 1.46 Support: 0.03
 {40} => {8} Confidence: 0.37 Lift: 1.46 Support: 0.03
 {5} => {43} Confidence: 0.20 Lift: 1.84 Support: 0.02
 {43} => {5} Confidence: 0.19 Lift: 1.84 Support: 0.02
 {6} => {43} Confidence: 0.19 Lift: 1.70 Support: 0.03
 {43} => {6} Confidence: 0.24 Lift: 1.70 Support: 0.03
 {8} => {43} Confidence: 0.19 Lift: 1.76 Support: 0.05
 {43} => {8} Confidence: 0.45 Lift: 1.76 Support: 0.05
 {6} => {5} Confidence: 0.21 Lift: 2.00 Support: 0.03
 {5} => {6} Confidence: 0.28 Lift: 2.00 Support: 0.03
 {8} => {5} Confidence: 0.17 Lift: 1.58 Support: 0.04
 {5} => {8} Confidence: 0.40 Lift: 1.58 Support: 0.04

{9} => {5} Confidence: 0.27 Lift: 2.57 Support: 0.02
 {5} => {9} Confidence: 0.19 Lift: 2.57 Support: 0.02
 {8} => {51} Confidence: 0.12 Lift: 1.25 Support: 0.03
 {51} => {8} Confidence: 0.32 Lift: 1.25 Support: 0.03
 {8} => {52} Confidence: 0.10 Lift: 1.52 Support: 0.03
 {52} => {8} Confidence: 0.39 Lift: 1.52 Support: 0.03
 {8} => {53} Confidence: 0.10 Lift: 0.97 Support: 0.02
 {53} => {8} Confidence: 0.25 Lift: 0.97 Support: 0.02
 {8} => {59} Confidence: 0.09 Lift: 1.50 Support: 0.02
 {59} => {8} Confidence: 0.38 Lift: 1.50 Support: 0.02
 {61} => {6} Confidence: 0.29 Lift: 2.07 Support: 0.02
 {6} => {61} Confidence: 0.15 Lift: 2.07 Support: 0.02
 {8} => {6} Confidence: 0.22 Lift: 1.57 Support: 0.06
 {6} => {8} Confidence: 0.40 Lift: 1.57 Support: 0.06
 {8} => {61} Confidence: 0.13 Lift: 1.76 Support: 0.03
 {61} => {8} Confidence: 0.45 Lift: 1.76 Support: 0.03
 {8} => {66} Confidence: 0.12 Lift: 1.85 Support: 0.03
 {66} => {8} Confidence: 0.47 Lift: 1.85 Support: 0.03
 {8} => {79} Confidence: 0.08 Lift: 1.66 Support: 0.02
 {79} => {8} Confidence: 0.42 Lift: 1.66 Support: 0.02
 {9} => {8} Confidence: 0.40 Lift: 1.56 Support: 0.03
 {8} => {9} Confidence: 0.12 Lift: 1.56 Support: 0.03
 {6}{8} => {12} Confidence: 0.40 Lift: 2.05 Support: 0.02
 {12}{8} => {6} Confidence: 0.30 Lift: 2.13 Support: 0.02
 {12}{6} => {8} Confidence: 0.51 Lift: 2.01 Support: 0.02
 {8} => {12}{6} Confidence: 0.09 Lift: 2.01 Support: 0.02
 {6} => {12}{8} Confidence: 0.16 Lift: 2.13 Support: 0.02
 {12} => {6}{8} Confidence: 0.12 Lift: 2.05 Support: 0.02

MAPREDUCE PROGRESS

Connected to 2 workers, MapReduce execution on the parallel pool

```

Editor - run.m
Parallel mapreduce execution on the parallel pool:
*****
*          MAPREDUCE PROGRESS          *
*****
Map    0% Reduce    0%
Map    5% Reduce    0%
Map   10% Reduce    0%
Map   15% Reduce    0%
Map   21% Reduce    0%
Map   26% Reduce    0%
Map   31% Reduce    0%
Map   36% Reduce    0%
Map   42% Reduce    0%
Map   47% Reduce    0%
Map   52% Reduce    0%
Map   57% Reduce    0%
Map   63% Reduce    0%
Map   68% Reduce    0%
Map   73% Reduce    0%
Map   78% Reduce    0%
Map   84% Reduce    0%
Map   89% Reduce    0%
Map   94% Reduce    0%
Map  100% Reduce   50%
fx Map  100% Reduce  100%

```