DATA MINING ENABLED MODELS FOR EFFICIENT DATA TRANSMISSION IN INTERNET OF THINGS

Thesis submitted to the Jawaharlal Nehru University

in partial fulfillment of the requirements

for the award of the degree of

DOCTOR OF PHILOSOPHY

IN

COMPUTER SCIENCE AND TECHNOLOGY

By

MUKESH KUMAR

SUPERVISOR

Dr. SUSHIL KUMAR

SCHOOL OF COMPUTER & SYSTEMS SCIENCES
JAWAHARLAL NEHRU UNIVERSITY
NEW DELHI – 110067, INDIA

June, 2022

School of Computer & Systems Sciences
जवाहरलाल नेहरू विश्वविद्यालय
JAWAHARLAL NEHRU UNIVERSITY
NEW DELHI-110067

CERTIFICATE

This is to certify that the thesis entitled **"Data Mining Enabled Models for Efficient Data Transmission in Internet of Things"**, being submitted by ***Mr. Mukesh Kumar*** to the **School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi**, in partial fulfillment of the requirement for the award of the **Degree of Doctor of Philosophy in Computer Science and Technology**, is a bonafide research work carried out by him under the guidance of ***Dr. Sushil Kumar***.

This research work embodied in the thesis is original and has not been submitted for the award of any other Degree.

June 23, 2022
Dr Sushil Kumar
(Supervisor),
Assistant Professor,
SC&SS, JNU,
New Delhi-110067

24-06-2022

Prof. T. V. Vijay Kumar
Dean, SC&SS
Jawaharlal Nehru University
New Delhi-110067

School of Computer & Systems Sciences
जवाहरलाल नेहरू विश्वविद्यालय
JAWAHARLAL NEHRU UNIVERSITY
NEW DELHI-110067

DECLARATION

This is to certify that the thesis entitled **"Data Mining Enabled Models For Efficient Data Transmission in Internet of Things"**, being submitted to the **School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi**, in partial fulfillment of the requirement for the award of the **Degree of Doctor of Philosophy in Computer Science and Technology**, is a bonafide research work carried out by me.

This research work embodied in the thesis is original and has not been submitted for the award of any other Degree.

Mukesh Kumar
97/10/MT/011
Ph.D Student
SC&SS, JNU
New Delhi-110067

ACKNOWLEDGEMENT

Working on research for my Ph. D at Jawaharlal Nehru University was both a rewarding and challenging experience for me. Many people contributed directly or indirectly to my doctor al research work over the years. I'd like to extend a small compliment to all of you.

I would like to express my heartfelt gratitude to my Supervisor, Dr. Sushil Kumar, for his invaluable guidance and outgoing personality, which enabled me to complete my doctoral research work. His upbeat attitude and valuable extra time aided me in resolving all of the issues that arose during the course of my research.

I would like to convey my deep regard to Prof. T. V. Vijay Kumar, Dean SC&SS for their wise counsel and indispensable advice that always encouraged me to work hard for completion of the thesis.

I am delighted for having Dr. Pankaj Kumar Kashyap, Miss Rinki Rani, and Dr. Ankita Jaiswal, as research collaborators who have provided a great company for the scientific and interesting discussion in my lab. Thank you for your never-ending patience and trust and numerous discussions of my ideas.

I would like to thanks to Mr. Kailash Chand. Sr. Technical Assistant for their help and encouragement and support.

I also thanks to all my friends who directly or indirectly have lent their helping hand in this venture.

My highest gratitude goes to my parent for their relentless supports, blessing and encouragement.

Finally, last but not least to thanks to my family, this work will not be completed without the confidence, strength and support of my family. My family has always been a source of motivation and encouragement. I would like to give my heartiest thanks to my daughter Bhavani and my son Chirag for their understanding and persistent love that empowered me to complete this work. Special mention goes to my beloved wife, Kiran Jeph for patiently supporting me throughout the writing process, who is always the source of inspiration for all my achievements. I am forever thankful to my wife, and my daughter and son to whom I dedicated this thesis.

Mukesh Kumar

TABLE OF CONTENTS

INDEX **PAGE NO.**

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

Abbreviations	Descriptions
AL	Application layer
ARIMA	Autoregressive-Integrated Moving Average
CE	Convergent Encryption
CBCF	Clustering Based Collaborative Algorithm
CDMA	Code-division multiple access
DM	Data Mining
DML	Data Mining layer
FCMCF	Fuzzy C-means Based Collaborative Algorithm
FDES	Fuzzy Descriptive Evaluation System (FDES)
GPS	Global Positioning System
GPRS	General Packet Radio Services
GIS	Geographical Information System
IoT	Internet of Things
IoV	Internet of Vehicles
IP	Address Protocol
IL	Information layer
IBCF	Item-Based Collaborative Filtering
KDD	knowledge discovery databases
KGC	Key Generation Certification
MANET	Mobile Ad Hoc Network
MATLAB	MATrix LABoratory
ML	Machine learning
MDP	Markov decision problem

POW	Proof of Ownership
RFID	Radio Frequency Identifications
RL	Reinforcement learning
SE	Searchable Encryption
TL	Transport layer
UBCF	user-based collaborative filtering
WSN	Wireless Sensor Network
XML	Extensible markup language

LIST OF PUBLICATIONS

Journals

1. Mukesh Kumar, Sushil Kumar , “Towards data mining in IoT cloud computing networks : Collaborative filtering based recommended system”, *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 24, Issue 5, pp. 1309-1326, Sept. 2021 (**Scopus Indexed**)

2. Mukesh Kumar, Sushil Kumar, “Q-Learning Enabled Green Communication in Internet of Things”, *Journal of Information Technology Management*, vol. 14, pp. 103-117, Jan. 2022 (**Scopus Indexed**)

3. Mukesh Kumar, Sushil Kumar, “Secure Data Deduplication and Sharing in Internet of Things”, Ad Hoc & Sensor Wireless Networks, (SCI Indexed) (Communicated)

Conferences

1. Mukesh Kumar, “'Q-Learning based Algorithm for Energy Balanced Routing in IoT", 2nd International Conference on Network and Cryptology (NETCRPT-2020), SC&SS, JNU, New Delhi, 4-6 Dec. 2020. (**Paper Presented**)

2. Mukesh Kumar, “Data Mining for the Internet of Things", 2nd International Conference on Network and Cryptology (NETCRPT-2020), SC&SS, JNU, New Delhi, 4-6 Dec. 2020. (**Paper Presented**)

ABSTRACT

It was like a dream, to connect everything on this earth for computation and communication, but the Internet of things (IoT) makes this “impossible” possible. With the introduction of the Internet of Things, devices with varying size and computational capabilities can be connected for communication. These communication devices produce data which is converted into knowledge for making decisions. The sensor devices deployed in IoT environment generate huge amount of data and all data are not useful. It occupies storage space which is not cost effective and energy efficient. Therefore useful information need to be extracted to make appropriate decisions. Extraction of knowledgeable data from raw data is possible with the help of data mining. Data mining for IoT is used to formulate an intelligent environment. Therefore, this thesis present data mining enabled models for data transmission for IoT, their applications, challenges in developing IoT environment and few open research issues.

The ever-increasing data due to gaining popularity of Internet of things (IoT) needs to be adequate storage compute complex task with accuracy and low latency and finally recommend the user’s preference in the near future. The cloud computing and data mining technique having ability to provide open platform for communication and generate precise recommendation. In this regard, this work mainly carried out two aspects: firstly, we build four layers IoT cloud computing architecture that provides an open platform for communication with various heterogeneous multi-source things. Secondly, we present a recommended system model based on collaborative filtering algorithm to enhance the accuracy rate of the items in the top priority recommended list. The proposed model inherently utilizes the user-item’s scoring matrix, asymmetrical influence degree on the

similar items between users and time weight decay function for the user's preferences. Finally, extensive simulations are done to show the accuracy rate, loss rate and recall rate of recommendation for the proposed model. Further, comparative analysis of results proved that our proposed model outperforms than other state-of-art model in terms of accuracy rate of the recommendation on the item with respect to data sample set size.

Limited energy capacity, physical distance between two nodes and the stochastic link quality are the major parameters in the selection of routing path in the internet of things network. To alleviate the problem of stochastic link quality as channel gain, reinforcement based Q-learning energy balanced routing is presented in this paper. Using above mentioned parameter an optimization problem has been formulated termed as reward or utility of network. Further, formulated optimization problem converted into Markov decision problem (MDP) and their state, value, action and reward function are described. Finally, a QRL algorithm is presented and their time complexity is analyses. To show the effectiveness of proposed QRL algorithm extensive simulation is performed in terms of convergence property, energy consumption, residual energy and reward with respect to state-of-art-algorithms.

With the rapid growth in the field of Internet of Things (IoT), a large amount of data is being generated by different IoT devices and stored at the cloud server. However, several users outsource the same data to the cloud server and waste a large amount of storage and backup spaces. Data deduplication is the most effective way to tackle this problem. But deduplication raises various issues like dynamic ownership, efficient key management, and security. Additionally, in the post-quantum era, traditional scheme will be vulnerable to quantum computers. Hence, we need to develop a safe, efficient, and cost-effective cloud storage system. In this context, we propose a fog-enabled secure data deduplication and encrypted search (FDES) scheme for IoT, which uses the Merkle tree for storage of data. The FDES scheme supports dynamic ownership management, and it provides authorized access to the shared data with quantum-resistant attacks using lattice cryptography. We further ensure a multi-user search for the encrypted data using an encrypted keyword based on the homomorphic function. Moreover, each authorized user in the system has his own

unique keys which further simplifies key revocation. Security analysis depicts that the performance of FDES scheme is better than the current data deduplication scheme. Moreover, simulation results demonstrate that encryption and decryption time is significantly reduced.

The proposed models/algorithms for data filtering, data transmission, and data sharing have been tested and validated by conducting simulations by writing own scripts in MATLAB. In general, the proposed models and algorithms presented in this work outperform the state of art models and algorithms in the literature.

Chapter 1

Introduction and Related Works

1.1. Introduction

Internet of things (IoT) is a popular research topic in technology, where different kinds of devices connect with each other through the internet. The devices are termed as smart objects/ things. These smart devices have the ability to sense the environmental conditions to make a decision as per the pre-defined constraints. It is a global platform to create a smart environment where things communicate, compute, coordinate and make decisions. Such an environment minimizes man power globally. Therefore, things are identified uniquely and automatically [1-3]. After a vast study on IoT, it came into view that IoT has many applications and standards followed with various challenges. Different surveys overview about five layered architecture of IoT to describe overall working design. The five layers are edge technology, access gateway, internet, middleware and application. To study IoT, there are three different angles such as the internet, things and semantics. The sensor nodes deployed to form an intelligent environment produce huge amount of data which occupies storage space and consumes lots of energy. All the data produced by sensor devices is not useful. Therefore it wastes the memory space and degrades the energy. As the devices are battery powered, so it is wise to utilize the available energy in an effective and efficient manner. The data stream in IoT systems in increasing continuously that is used to develop business models, customized products enabled with personalized services [4-6].

The infrastructure is well understood, but the question is how the produced signals or data can be formed into knowledgeable data. The answer can be data mining (DM). Data mining is helpful in finding a solution for this issue of extracting important data from raw data. Data mining is the technology which extracts hidden information from raw data. This is known as knowledge discovery in databases. The integration of KDD and DM facilitates to generate highly intelligent and operational systems. Data mining has various technologies to extract useful information. All the available technologies are application dependent. A vast research has been done to develop data mining technologies in IoT to strengthen the performance of smart environments. It makes IoT smarter with intelligent services. This chapter elaborates a detailed study of data mining techniques for IoT [10-11].

This chapter is organized as: section 1.2 is explores the related work to describe applications of IoT based system and various data mining techniques for the same. Section 1.3 elaborates rules for selecting DM techniques in different IoT environment. A comprehensive comparison of DM algorithms is shown in tabular form, followed with challenges in developing such environment with DM techniques in section 1.4. Section 5 present the problem statement and objective of the thesis. Section 1.6 provides the origination of the thesis.

1.2. Background

This section addressed the applications of smart environment followed with introduction of DM techniques in IoT system.

1.2.1 Applications of IoT based system

Healthcare applications: Now a day, the medical equipment in hospitals are trained and intelligent equipment that carries health data of thousands of patients which provides essential additional information regarding the disease. Such information provides further treatment interventions and potential preventive measures in different cases [7].

Monitoring patient remotely: Health care professionals, family members and other professionals care takers involved in treatment can monitor and optimize real time changes in patient's health when they are not able to reach at location. This application reduces the need of medical professionals and other medical equipment significantly. This application Benefits in taking decisions for any critical situation with different standard opinions. Such monitoring techniques benefits the patients whose routine health monitoring is mandatory. This is a preventative and early diagnose technique [8].

Monitoring remote locations: This application includes similar features of remote patient monitoring. Remote locations or out of human reach locations benefitted with this application to monitor various events remotely [8].

Monitoring assistive equipment: Assistive equipment are deployed in smart homes or hospitals especially for disabled and elderly peoples to enhance the quality of life. Such applications are life changing advancements in technology. Examples for such assistive equipment are smart wheelchairs, wheelchair management systems etc. to monitor the status and location of users. Smart homes are equipped with such devices to control the accessibility and enhance the quality of life. Such devices are capable of collecting potential information which defines the basic routine of user [9].

Traffic monitoring: Vehicular traffic or live traffic can be monitored before the worst situation and action can be taken accordingly. The status of path in near future during a drive can be monitored and alternate way can be generated or shown [9].

The smart objects used in IoT environments generate massive information, which is used in various applications. The information from smart devices is extracted to convert it into useful information. Extraction process eliminates garbage information from raw data and extracts hidden information. This can be done through data mining algorithms. The IoT platform is increasing at a very fast rate, hence the smart devices produce large amounts of data. The large data is termed as big data which needs to be analyzed to make it useful at its maximum capacity. Data mining discovers novel, useful, and interesting information patterns from a set of large data. Then it applies a data mining algorithm on it to extract hidden information from it. Knowledge extraction, knowledge discovery databases (KDD), data archeology, data pattern analysis, information harvesting etc. are few terms used in data mining (DM). DM process works in developing an effective and efficient model, capable of generalizing new data, discovers specific information from a large set of databases, data warehouse and from other data repositories. The DM process includes few steps such as data preparation, data mining and data presentation. Data preparation steps make data to be prepared which consist of three sub steps such as integrating data from various sources, cleaning of noise from data and making it ready for pre-processing. To evaluate or find useful patterns from the collected data in order to classify data for knowledge discovery, DM steps are followed. Data presentation step makes the extracted data presentable for the viewer [10].

1.2.2 Data Mining Techniques

The data around us is useless until it is processed under data mining techniques. With the utilization of DM techniques, the IoT environment becomes intellectual. For automatic data analysis, data mining techniques are divided as supervised, unsupervised and reinforcement learning. The analysis of data under DM techniques provides more precise results as it goes through multiple layers. Supervised and unsupervised learning together for automatic data extraction is also considered as machine learning techniques. The raw data is collected from various IoT devices which is further forwarded for the knowledge discovery process. In knowledge discovery, data is pre-processed to mold raw data into a relevant format for analysis. Under pre-processing, various actions are performed such as feature selection, extraction (eliminate garbage information), noise abstraction, normalization dimension reduction are performed. After preprocessing the data goes under a data mining process where pattern discovery, recognition, abstraction, filtering and event sequence detection are performed. Data pre-processing and data mining are together known as deep learning [24-25]. After DM techniques, the data is utilized for decision making, automation and optimization purposes by the IoT infrastructure. Data mining techniques are divided into four broad categories such as classification. Clustering, association rules and frequent pattern discovery method [11], [20].

1.2.2.1 Classification

Categorizing the available data with respect to some predefined targets is known as classification of data. Its main goal is to forecast the target class for accuracy. Classification is a type of supervised learning process because target labels are supposed to be known before the pre-processing. The classifier or prediction function requires training, so as to classify unlabeled data. The labelled data is used to train the classifier. In the initial stage, the classifier is built from a set of rules by previously available data. The data can be labelled or unlabeled. The labelled data is also known as the training set of data and unlabeled data is also known as testing set of data. The classifier is first constructed by training data then validation is done through testing data followed with analysis of data to classify the data in an appropriate class. Classification algorithm computes the probability of relevance of an item to a particular class, then compares the cutoff value. The performance of classification is computed by evaluating accuracy level and error rate. To classify the data, decision tree induction, neural networks, Bayesian network, support vector machine, rule based classification, classification by backpropagation, deep neural network, frame based and ensemble methods are used [40]. For large scale complex applications, fusion of different classification techniques are adopted [21]. The most suitable classification methods for today's IoT environment are rule based, support vector machine and association based analysis. An intellectual model can be developed using the hidden Markov model of data mining. In biomedical and smart city applications naïve Bayes, Gaussian naïve Bayes, Bayesian belief network, artificial neural network and ensemble method are most suitable [12-13] and [51].

1.2.2.2 Clustering

Dividing the data into meaningful groups is known as clustering. The data in a group constitute similar features. Clustering is an unsupervised learning process because it does not require prior knowledge to group the data into clusters. Various clustering techniques are hierarchical clustering, partitioning algorithm, co-occurrence, scalable high dimensional clustering, K-means, K-nearest neighbor, K-medoids, grid based clustering etc. In a smart IoT based environment, cloud based distributed clustering is more suitable than centralized clustering. Data in cloud based distributed clustering are accessible by everyone. This feature has its own pros and cons. This face privacy issues [14-19].

1.2.2.3 Association rule or frequent pattern mining

A set of objects appears repeatedly are known as frequent patterns. In felicitous environment, frequent pattern mining provides better analytical understanding. Association rule helps in predicting an accurate pattern of an event. Mining the relevant frequent pattern are also known as sequential pattern mining. Sequential pattern mining is more attractive than frequent pattern mining which can analyze a sequence of event in a particular time frame. It is used for event discovery and event recognition. To diagnose the occurrence of an event can be measured through frequent pattern mining. For ex. In the medical field, such mining is used to diagnose the early occurrence of a disease. It observes the gradual internal changes in the body and extracts some useful pattern that might be the initial symptoms of a big disease. It observes the deviation in patients' health. Support and confidence are the two terms used to predict the early signs of a disease [35-39]. Various association rules for frequent pattern mining are Boolean association rule (a priori knowledge based algorithm), class sequential rule mining and clustBigFIM [48-50].

The unforeseen useful information from raw data is called anomalous objects or outliers. Outliers are different from regular data objects and provides interesting inherent features. Outlier deviations are very useful in IoT applications such as smart home, smart traffic, smart agriculture and packing systems. There are four attractive outlier approaches named as statistical distribution based outlier detection, distance based outlier detection, density based local outlier detection and deviation based outlier detection [44]. In a study provided by Bishwas and mishra in 2015, an IoT based environment is developed to monitor health where a bio- metric sensor and Arduino UNO based setup is developed to monitor health parameters [22-23]. After this outlier detection mining is applied to extract anomalous information for an emergency like situation. This is a cluster based analysis framework with recursive principal component analysis to enhance the effectiveness of system. The outlier approach achieves fast convergence [32-33].

This is concluded after studying various DM techniques that an algorithm must include multi- dimensional parallel real time stream processing, time scan, multilevel and analysis ability to improve the effectiveness and convergence rate. All the techniques are application dependent, therefore applicability of various techniques varies. The comprehensive comparison of various DM algorithms is addressed in table 1.1.

1.3. Data Mining Techniques for IoT

This section elaborates about the most applicable data mining technology for IoT which is suitable for development of high performance systems. The data mining algorithms such as classification, clustering, association rules and frequent pattern methods already discussed earlier in detail. . To describe IoT, two simple phrases are used: " data about things" and" data generated by things" that refers to data to define things and data captured from sensor devices also refer to "big data". The amount of big data is around zettabyte which cannot be handled with the traditional data analysis tools. Single storage systems cannot store zettabytes of data. Therefore the traditional tools are not able to process and analyses big data. There are few traditional data mining methods such as divide and conquer, random sampling, incremental learning, and data condensation that can handle and analyze big data produced from IoT devices [45][47]. These methods capture interesting patterns from sensor devices to reduce the complexity of input data. Few traditional data mining methods are able to reduce patterns as well as number of dimensions in order to improve the convergence rate. Such methods are helpful in developing applications like smart homes or smart cities. As discussed in the previous section, KDD is successfully applied to extract hidden information from raw data with the following steps: selection, pre-processing, transformation, DM and interpretation. Data pre- processing includes selection, pre-processing and transformation, decision making includes interpretation and evaluation steps. Data pre- processing steps are taken before DM steps and decision making steps are taken after DM steps. Data mining steps are responsible for data extraction from output of data processing and forward then for decision making steps, where transformation is done. It is understood that all the attributes are not useful for mining, therefore the selection step selects key attributes.

Various data mining algorithms are used to enhance the intelligence of the IoT system which is used to foresee the actions of occupants in a smart environment. The data mining technologies are not restrictedup to smart environments but proves their efficiency in other domains well. Various studies for the relevant topics provide the successful use of data mining technologies in smart or self-intelligent environments. It enhances the smartness of provided IoT infrastructure. The example for smart infrastructure are event detection spots, smart supermarkets, traffic management and various transportation management systems. Such infrastructure improves the overall performance of activities. Data sources for such infrastructure can be deployed as sensor devices. Data mining technologies extracts the interesting patterns of information from sensor devices. To extract important information, metaheuristic algorithms are also used which provides optimized solutions.

There are few rules on the basis of which these DM algorithms are adopted in different IoT environments. The rules can be adopted as per the rules discussed below:

Rule 1. Divide the DM technologies into two classes depending upon the characteristics of the problem. Classification and clustering into one class and association and frequent pattern in another class.

Rule 2. Classification works more suitable with labelled data as well as unlabeled data while clustering works for unlabeled data only. Therefore, further divide the problem accordingly.

Rule 3. Frequent pattern method works when the data is in a particular sequence while association deals with a set of relevance of data. There is no particular order of data in association rule based events. Therefore, further decide the problem accordingly.

All the above mentioned DM technologies have their own pros and cons. As the need of IoT environment is changing with respect to modernization of living style [30-31].

However, single DM technique won't work effectively in large scale smart environment. Therefore a combination of DM technologies at different levels are adopted for better results. For example: (a) clustering and classification are combined to work as an unsupervised learning system, where an automatic set of classifiers is generated through clustering without any prior knowledge of input patterns, then incoming patterns are classified through classification methods. (b) Classification and clustering are combined to make another integrated system where classifiers are generated through classification methods from known dataset and new classifiers are added to existing classifiers through clustering. This combination acts as semi- supervised learning methods. These methods are capable of handling data from IoT dynamically. (c) Clustering, classification and frequent pattern methods forms another combination to make a single system for analysis of information. The three DM technologies can be arranged in different orders to make a new system depending upon the requirement of the problem statement. In the same way clustering, classification and association rules are also arranged in different-2 orders to make a single system for data analysis. These systems can perform their tasks repeatedly in a loop to create better solutions. Such systems can be viewed in smart health analysis tools, smart homes, smart cities etc.

Table1.1:*Different data mining algorithms for IoT systems*

Data Mining Algorithms	**Techniques**	**Objective**	**Source of Raw Data**
Classification	KNN, Naïve Bayes, Logistic regression, Support Vector Machine [34]	Event detection, traffic management, parking management, Action discovery, recognition, identification and prediction	Text data, Sensor data signals, video camera, microphone, Smart meters, Smart energy devices, smart phone, wearable sensor devices, smart health care devices and machineries
Clustering	K-means [42], K-anonymity, Micro aggregation. Extended finite automation	Performance measurement, enhance quality of life, energy preservation, security and privacy	
Association Rule	Residual method, Unsupervised and probabilistic IPCL data fusion technique	Relevant action prediction	
Frequent Pattern	Fp-growth, Episode discovery sequential pattern mining [43],	Tag management for RFID [29] [43], event behavior analysis, pattern recognition	

	Unsupervised discontinuous varied-order sequential miner		

1.4. Challenges with Data mining for IoT

This is understood that without data mining technologies, the dream of a smart environment is just a dream. But data mining technologies along with cloud computing techniques, makes this dream more applicable. The demand of society from technology is increasing day by day, so with traditional DM techniques, it is not possible to connect everything and computing information effectively. After all the mentioned applications there are few open issues with DM techniques for IoT systems. These issues are related to scalability of big datasets.

1.4.1 Issues with infrastructure

The decentralized and heterogeneous nature of IoT system affects DM techniques. Smart environments must support decentralized data storage and computing capabilities. Existing smart environments have centralized systems for computing and storage systems which need to be decentralized for better performance. Centralized system consumes more energy. It is observed that decentralization is not required in all the situations of IoT, but a completely centralized system increases the energy consumption level. Sometimes the performance of an IoT system is degraded due to not having accessibility to all the data. The existing DM technologies are designed for small scale smart applications; therefore, they provide low computation and low throughout when applied to large scale infrastructure. For large scale infrastructure, some cloud based systems should be developed. But cloud based systems face different challenges in terms of cost of computing.

1.4.2 Issues with data

Data preprocessing, information extraction and retrieval are the three ways to deal with large scale data produced from IoT devices. The sensor devices have limited size of memory, therefore redundant data and unimportant data need to be eliminated from the storage space of sensor devices in order to upgrade overall execution of the system. The solutions such as dimension reduction, data compression and data sampling are adopted. Acquisition, deposition, analysis and integration employed various issues that affect the performance of the system. However, there are several standard protocols which define the connectivity parameters of different sensor devices, so as to make the produced information useful. But the meaning of input produced from heterogeneous sensor devices is not the same in all the applications. This issue can be tackled through few technologies such as ontology, semantic web, extensible markup language (XML) etc [41]. But these technologies are not appropriate to produce final solutions.

1.4.3 Issues with algorithms

For a complex smart application environment, it is very difficult to select DM technologies for integration. The order and selection of DM modules to design an optimal solution for extraction of useful information is also challenging. Overfitting is another issue under algorithm issue. This can be explained as, the labelled data are used to train the classifier and labelling of data is very expensive. The labelled data is also known as training patterns. More the training patterns, higher the accuracy rate. The balance between cost and accuracy of a model is challenging for selection of an algorithm.

1.4.4 Issues with privacy and security

A promising paradigm, IoT with numerous applications in different domains, faces security and privacy issues. The applications like smart meters, monitoring patient remotely, smart cities, waste management and industrial controller's demands security for their data, which is violating in current scenario. The security of data is an essential requirement of personal data such as living patterns, habits, preferences and social requirements [25-28]. Massey, Anton et al explains a framework to observe the privacy policies of the system. It explains the concept for using the suitable positions to apply privacy policies and examined about the input and output security concerns. It is a five stage policy framework for security and privacy of smart environment. Evan and Eyers et al. [55] examined about the usage of tagging techniques for the sake of privacy which also helps the system to under the flow of information. But this methodology is expensive in terms of processing, storage and communication. A trust based model for privacy preservation is developed by Appavoo, chan et al. [56] with an objective to improve privacy. This methodology ensures and restricts the unauthorizeduser's accessibility. Anonymization algorithms produced by Otgonbayar, Pervez et al. Which supports k-anonymity privacy model[57]. It examines the similar input and groups them into clusters and utilizes a time- based sliding window technique for anonymized the input data. It supports rapid cluster formation. The model is validated on real time dataset and proves its effectiveness with minimum information loss and higher convergence rate. Cryptographic techniques also shows a great privacy concerns, the model is proposed by two, Alelaiwiet al.[58]. Although, the model is expensive but very effective in real time medical domains. It is a multi- party framework which hides the data from attackers. In this type of model, every sensor node contains a private secret key to ensure the privacy of the information shared between the devices [46]. A sensitive and privacy based environment framework is proposed by Perez et al. for health care and automation systems applications [59]. This approach is also based on cryptographic techniques. This model ensures the secure exchange of data between devices and handles protected data. A modern privacy framework addressed by Ge, Hong et al. [60] To tackle new security issues which incorporates different phases such as data processing, security model generation, visualization, analysis and model updates. This is applied on IoT generator nodes, security model generator and security evaluator. This model is potential security defender. Computerized numerical control information based privacy mechanism addressed by Li and Li which is a lightweight authentication method for security of information based on organizational characteristics in IoTenvironment [61]. The protocol includes five parts like system setup phase, sensor node registration phase, user node registration phase, login phase, authentication and session key agreement phase. A series of analysis id done to prove the efficiency of secure environment. This protocol employee doubles privacy protection strategy [52-53].

1.4.5 Networking issues

When devices are connected to share data, there must be appropriate signal quality in order to protect the data integrity. The devices in IoT system are different in terms of scalability and computing capability, they might generate data in different way, and therefore connectivity between these devices may cause an issue. Connect between different devices depends on various factors such as availability, interoperability, cost, scalability, reliability, coverage, data rate and power consumption. The sensor devices are connected through low range wireless communication network and gathered data is forwarded through large scale wireless communication network. Robust and scalable connection is mandatory requirement for IoT network. The type of network selection is based upon type of application to maintain proper network connection. This issue should be taken care on priority basis [54].

1.5. Problem Statement and Objectives

As the sensor devices produce a huge amount of data and sensor devices are battery powered, therefore an energy efficient mechanism should be developed with some improvement scopes with the passage of time. Such environment will always be in the scope of improvements. That's why energy issue in IoT environments will always be a hot topic of research. Along with energy, the huge amount of produced data may suffer from congestion issues. Therefore energy efficient data transmission could be the sensitivity research topic in IoT environment. Further, the sensor devices have a limited storage capacity and this is an era of technology, therefore cloud based or fog based mechanism should be developed with some advanced features to incorporate storage concept. Security and privacy is always a prime concern for any technology in order to prevent data from cyber-attacks or unauthorized access. To provide the solution of the above said the problem, following objectives have been considered.

- To develop IoT cloud computing architecture that provides an open platform for for data management, storage, analysis and uses data mining approach to recommend top N- priority list to the target users.

- To develop collaborative filtering assisted to recommender system model to improve the accuracy rate of the items in the top priority recommended list.
- To design energy balanced routing algorithm based on reinforcement learning approach that include residual energy, physical distance and link quality of the channel for data transmission.
- To develop a safe, efficient, and cost-effective cloud storage system that provides secure data deduplication and data sharing in IoT and cloud environment.
- To compare the proposed models with the existing state art of models.

1.6. Organization of the Thesis

The rest of the thesis is organized as follows. In chapter 2, a collaborative filtering based recommended system towards data mining in IoT cloud computing networks is presented. In chapter 3, a Q-Learning Enabled Energy Efficient Data Transmission scheme in IoT is presented. Chapter 4 presents a fog-enabled secure data deduplication and encrypted search (FDES) scheme for IoT, which uses the Merkle tree for storage of data. Finally, chapter 5 concludes the work presented in this thesis with its scope for future enhancement.

Chapter 2

Towards Data Mining in IoT Cloud Computing Networks: Collaborative Filtering based Recommended System

This chapter primarily focused on two aspects: First, we create a four-layer IoT cloud computing architecture that serves as an open platform to communicate with a variety of diverse multi-source things. Second, we reveal a recommended system model based on a collaborative filtering algorithm to improve the accuracy rate of the top priority recommended list items. The proposed model inherently utilizes the user-item's scoring matrix, asymmetrical influence degree on the similar items between users and time weight decay function for the user's preferences. Finally, extensive simulations are done to show the accuracy rate, loss rate and recall rate of recommendation for the proposed model. Further, comparative analysis of results proved that our proposed model outperforms than other state-of-art model in terms of accuracy rate of the recommendation on the item with respect to data sample set size.

2.1. Introduction

Internet of Things (IoT) technology boom in the number of devices to connect the internet and in turn there is exponentially growth in information [62-63]. IoT build a network by connecting each ubiquitous physical thing (smart sensor chip) through radio frequency identifications (RFID) technology, global positioning system (GPS), infrared sensors technology or geographical information system (GIS) to internet for exchange of information throughout the globe [64-65]. These information's is further increases to astronomical figure as used by the Internet of enterprises through data mining technique subject to better portrait needs and satisfaction of user [66]. To connect these multi-source devices, storing the huge amounts of data, data packet acquisition and their analysis is the challenging issue. Cloud computing emerges as distributed platform to solve the problem of communication, computing and storage capacity, that integrates multi-source devices and abstract the interested information to the IT users [66]. There is still major problem is how to find the desired information and recommend to the target user from the huge amount of data. Data mining is the come up as prominent solution to solve the processing of big data, but it can be done on high-performance computers. It is further complex by high-dimensionality and unstructured information. Thus, in this chapter we proposed a novel IoT cloud system architecture to solve the problem of user management, communication, data storage, analysis, and huge data application visualize on user's side etc. we also proposed novel collaborative filtering algorithm to recommend the desired information (items) to the target user using similar neighbour's user scoring the items.

The IoT cloud integrated platform responsible for data gathering, data pre-processing, data exchange and must be specific business application [67-68]. Data cleaning and filtering technique is well defined in the chapter [69], where XML data cleaning is done through tree editing distance. In the chapter [70], XML data is filtered using Bayesian approach in XML Dup system. For data mining of the retrieved from the multi-source sensors, general architecture of IoT cloud computing is proposed in the literature [71-73]. Thus, it is most important to build an architecture that support data mining technology flawless for the massive amounts of data and generate accurate recommendation list of top priority items.

In the IoT cloud platform data mining approach for the discovery of available data from the retrieval of heterogeneous data by multi-source devices plays an important role [74]. This mass data mining technique is different than traditional data mining approach in the decision making for the recommendation of information in the upper layer application specific user. It must focus the specific data cleaning or filtering process according to application, classification; track the frequent patterns (correlation) in the similar data, preventive measures for IoT remote platform and other aspects [75]. There are two major aspects to cover the information overload for the energy and time saving of the users to select an item from the big data. One is search engine optimization, which provides solution to the target-specific data retrieval problem [76]. It fails in the case of when the needs of the user are not precise, because in some cases user is not clear about of its own needs. On the other hand, recommended system portraits the information to users according to their personalized information retrieval using data mining approach [77]. The recommended systems are mainly classified into four groups, content-based filtering, knowledge-based recommendation, collaborative filtering recommendation and combination recommendation [78-79].

In this chapter, collaborative filtering recommended system based algorithm is proposed to generate accurate information to the target user. It uses the rating or scoring feature of the users on the items (projects) to discover the potential needs of users. Collaborating filtering technique is not depending on the knowledge of the items or their analysis of content technology. Thus, collaborative filtering recommended algorithm has strong ability to adapt different environment and application. But, it tends to losses its predictive performance with the increase of sparsity in data [77]. Data sparsity problem arises, when there is sufficient rating (feedback) is not available to finding the others similar kind of users or neighbours. This is because of small number of rating is done by active users in the network. Which is further complex by cold-start problem i.e., generate recommendation to those users who rated only small number of items [80]. Thus, data sparsity in the recommended system pulls out more attention towards researches and academia. Content-based filtering, clustering, filling, dimensionality reduction are some of the major approaches to reduce the effect of data sparsity.

In this regard, we present system architecture of IoT cloud computing platform for data management, storage, analysis and uses data mining approach to recommend top N- priority list to the target users. Collaborative filtering algorithm is presented to improve the prediction accuracy on the basis of user scoring matrix. We also consider the asymmetrical influence degree of the scoring of user on similar items and time decay weight function for the recommended system. Finally, the simulated results show that accuracy rate of the proposed model is improved and loss rate is decreases as the data sample set size increases. Further, comparison of the proposed model with state-of-art-models is also presented in terms of accuracy rate.

2.2. Background

In recent years, numerous researches and academicians has have proposed data mining model based on simple filling. Collaborative filtering algorithm, dimensionality reduction, incorporative time decay function to find the distribution or recommend the items at time instant according to user needs. In the chapter [81], authors has used simple filling approach to fill out the user's evaluation (raring) on the certain items using unified numerical value and able to overcome the sparsity up to certain extent. where as in the chapter [82], authors have used clustering approach to classify the items in similar group on the basis of scoring information. The results of the proposed model show that accuracy rate of predicted recommended item is improved and able to deal with sparse data. To extent the clustering approach through fuzzy k-means algorithm, another model in the chapter [83] has been proposed by the same authors [82] to deal the scalabilities of users and data sparsity. Dimensionality reduction approach is also used to reduce the effect of absent scoring of users but it losses some user score's in the scoring-matrix. Above, mentioned model has good performance in the case of small-database application but performance of predication is decreases in the case of big data. This is because of in the simple filling approach prediction model needs to fill out the sparser rating of items or classification the user preferences to show their relative difference is not accurate in the case of clustering approach for big database.

In decision support system, Data mining technique give breakthrough in the analysing the expert experiences and building knowledge based discovery system for the prediction of user's information in the various field such as medical filed [84], urban traffic [85], education field [86]and so on. The decision making activities are trained using decision tree algorithm in data mining by solving the auxiliary probability of decision support system [87]. Decision tree algorithms such as CART, C5.0 and CHAID also take care of the effect on the system after taking decision. These algorithms can recommend the information according to user's needs in small application but fail to compile in big database application. This is because of size of decision tree increases more rapidly and taking a decision consumes lots of energy and time, which is unfavourable to IoT cloud computing environment.

Collaborative filtering algorithm can be modelled as user-based collaborative filtering (UBCF) [88] or item-based collaborative filtering (IBCF) [89]. In the chapter [90], authors has address the problem of Matthew effect, where list rated item become less popular on increasing time period and proposed a serendipitous innovator CF based recommended system to solve the cold-start problem. In the chapter [86], authors have proposed a novel cross-layer collaborative filtering model for the accurate prediction of score for the optional course using most senior student's scoring rate on the optional course. Whereas in the chapter [91] [92], authors have proposed clustering based and fuzzy C-means based collaborative algorithm (CBCF or FCMCF) respectively. In CBCF the accuracy rate of prediction and F1 score was improved by giving incentive or penalty to user according to their rating. Whereas in FCMCF address the sparsity concern using sparsest sub-graph detection algorithm and results shows that recommendation quality and adaptability of fuzzy logic in new environment is improved.

From the above mentioned literature, it is appeared that collaborative filtering algorithm emerges as viable technique, which predicts the user's preference more accurate than other data mining technique. In this chapter, we present a recommended system using collaborative filtering algorithm to predict user's preference more accurate using historical rating given by other active neighbours refer to user-items scoring matrix, time weight decay function.

2.3. System model

An intelligent cloud computing open platform for providing communication link between IoT devices and different technology for data mining is shown in the fig 2.1. The presented system architecture divided into four layers such as Information layer (IL), Transport layer (TL), Data Mining layer (DML) and Application layer (AL). An information layer collects the data from different users to data acquisition for the data optimization or mining in the real world IoT cloud computing network. These data may be obtained from GIS map, GPS statistical data of vehicles, wireless data or any other business information, which work as the base of data optimization. The TL is responsible for the communication between multiple processes running on the different host. For this purpose, transport layer uses its sharing library function and reuse function for data uploading in the data management centre through GPRS/CDMA, wireless RFID or Ethernet channel.

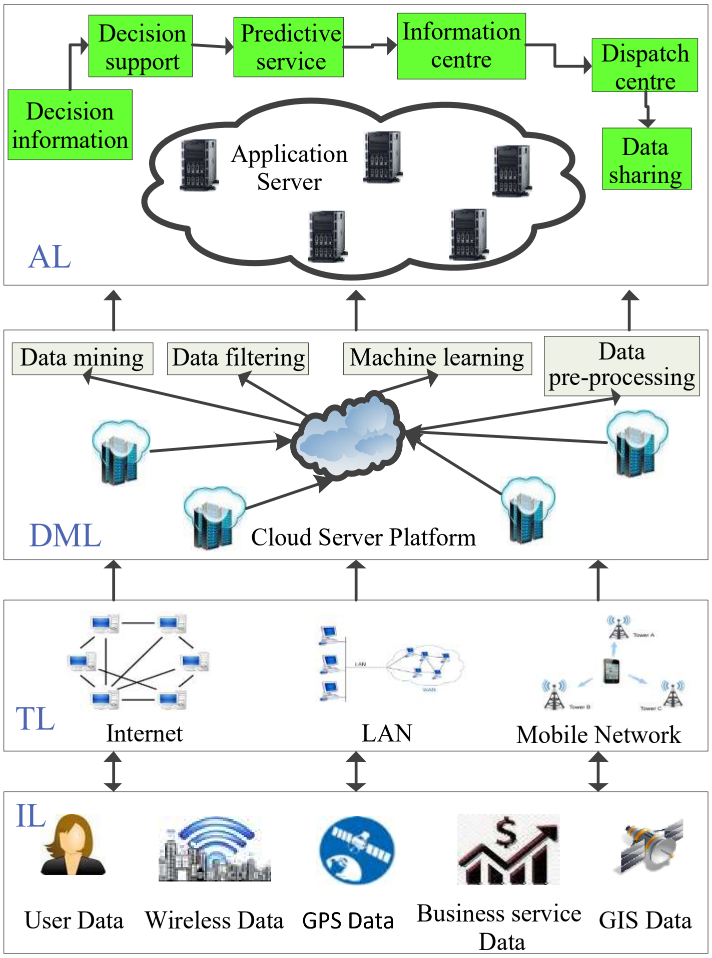


Fig.2.1. Four-layer architecture design for data mining in IoT cloud computing network

The DML is also labelled as bridge layer is the most important for providing services to the users according to their interest. The data mining techniques depends on the characteristic of IoT data obtained from the data management centre. This is because of there is heterogeneity, complexity, correlation and sparsity among the data. So, choosing the machine learning approach is far better than traditional approach. The DML firstly analzye the data characteristics, the proposed a appropriate solutions and then suggest a physical model for implementation. It use machine learning technology to analyse the each data and recommend the information to users for their visualization according to users rating to similar projects or search engine optimization techniques. Finally, the AL provide the services to users for business processing through communicating different application process running on different hosts. It provides visualization layer to the users, which is basically, relies on key information extracted from Data mining layer.

2.4. Collaborative Filtering recommended system of Big Data in IoT Cloud

In this chapter, we formulate the problem Top-N recommended information is visualizing to the users according to the neighbours users rating on similar projects or items based on Collaborative filtering. The primary concern to use the collaborative filtering is data sparsity under Big data in IoT cloud computing network. The data received from the IoT device of different characteristics such as relevance, poor quality, spatiotemporal, large amount of information (mass) and non-structural known as data sparsity. We propose a model based on user preference technique, asymmetric influence metrics among users on similar items and calculate the time weight function to predict the preference score for the recommended item for the user. The working steps of the presented model as follow:

2.4.1 User preference Technique

Table2.1. Scoring Matrix for user-items

Items→	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9
Users↓									
u_1	2	2	1	5	0	4	6	3	3
u_2	3	6	0	0	3	2	1	2	4
u_3	6	5	4	6	3	2	5	2	0
u_4	6	0	0	5	3	4	1	1	3
u_5	3	2	3	4	6	2	0	3	4

The core of the collaborative filtering recommended system is the building a scoring matrix i.e., users rate the items on the scale 0 to 6, being 0 shows that user is not interested in the item and 6 rating sows that user has more satisfaction on the item. Let us assume that number of users$u_p = \{u_1,\ u_2, \ldots u_i \ldots . u_p\}$ and items are$i_k = \{i_1,$ $i_2 \ldots i_k\}$, converting into scoring matrix (u, i) shown in table1.1. For example, we consider number of neighbour users is 5 for the target user and total number of items is 9.

From the table2.1, we observe that user$u_1$is more interested in item i_7 than useru_5, so it is clearly observed thatu_1 more enjoys the item rather than u_5 to item i_7 . Thus, it is necessary to calculate the user preference score (upt) for the item $i_k$similar to [74] as follow:

$$upt(u_i) = \delta . \frac{upt|u_i|}{n_u} + \mu . \sum_{u_j \epsilon \{u_1, u_2, \ldots u_i \ldots . u_p\}} \frac{upt\|u_j\|}{n_u} \tag{2.1}$$

Where, n_u define the number of rating given by user u, $upt\left|u_{i/j}\right|$ refers to number of rating given by user$u_{i/j}$. The weight of the Eq.(2.1) δ and μ control the normalized value and it is obtained by using chapter[74]. [86]. Now, we calculate correlationφ between users u_i and $u_j$using Pearson coefficient of similarity for finding the neighbour of concerned user as follow:

$$\varphi(u_i, u_j) = \frac{\sum_{i \epsilon\ u_i, u_j} \left(s(u_i)_{i_k} - \overline{s(u_\imath)}\right)\left(s(u_j)_{i_k} - \overline{s(u_\jmath)}\right)}{\sqrt{\sum_{i \epsilon I_{u_i, u_j}} \left(s(u_i)_{i_k} - \overline{s(u_\imath)}\right)^2} \left(s(u_j)_{i_k} - \overline{s(u_\jmath)}\right)^2} \tag{2.2}$$

Where, $I_{u_i, u_j}$is the set of items on which users u_i and u_j project same score$s(u_i)_{i_k}$ represent the score put by the user u_i and $\overline{s(u_\imath)}$ represent the average score of target user data, which is calculated byuseru_i.Now, the degree of interest $D(u_t)_{i_c}$ on the candidate itemi_c for the target useru_t is calculated as follow:

$$D(u_t)_{i_c} = \sum_{u \epsilon u_n} \varphi(u_i, u_j) \times \left(s(u_i)_{i_k} - \overline{s(u_\imath)}\right) + \overline{s(u_\jmath)} \tag{2.3}$$

Where, $u\epsilon u_n$represent the set of nearest neighbour of the target users. Using user reference technique, the above Eq. (2.3) can be appropriately converted as follow:

$$D(u_t)_{i_c} = \sum_{u\epsilon\ _n} \varphi(u_i, u_j) \times upt\big(D(u_j)_{i_c}\big) \tag{2.4}$$

2.4.2 Asymmetrical influence degree

As there is large number of heterogeneous users exit in the IoT cloud network, so there degree of influence on the similar item also varies greatly. The degree of asymmetrical influence$(\omega)$of the user u_i on user u_j is calculated as follow:

$$\omega\big(u_i, u_j\big) = \frac{1}{e^{\left(\frac{|I(u_i)\cup I(u_j)|}{|I(u_i)|}-1\right)}} \tag{2.5}$$

Where, $I(u_i)\ \epsilon\ [0,1]$ represent the influence degree of user u_i.

2.4.3 Time weight decay function on item

As, we know user interest on the item is varies along with time, their score for the selected item I in past can be changed in present or it can be decreases with time. So we calculate the influence of the time on the item as decay function can be given as:

$$\Phi\big(T(u_i)_{i_k}\big) = 1/e^{\left[\rho.(t_0 - T(u_i)_{i_k})\right]} \tag{2.6}$$

Where,Φ refer to time weight decay function,$T(u_i)_{i_k}$ represent the time at which user u_i score the item i_k in the past, t_0 is the time at which user's score is sorted out in the data management center and ρ is the weight of time decay function. It is clearly observed from the above Eq. (2.6), as there is difference in time interval larger, lesser the influence of the user's rating on the item.

2.4.4 Predicted score on the preference for target user on candidate item

User's scoring criteria for the item is different for different user. And, also the above two parameter define in Eq. (2.5) and Eq. (2.6) have impact on the predicted score on the preference for target user on candidate item, so final value of predicted score for the target user is calculated converting the Eq.(2.4) as follow :

$$D(u_t)_{i_c} = \sum_{u\epsilon\ n} \varphi(u_i, u_j) \times upt\left(D(u_j)_{i_c}\right) \times \omega\left(u_i, u_j\right) \times \Phi\left(T(u_i)_{i_k}\right) \qquad (2.7)$$

2.4.5 Proposed Collaborative Filtering Algorithm for the recommendation of preferred item for the target user

Algorithm 1-ProposedCFA for the recommendation to user

```
1. Input: $u_p$, target user $u_k$, user-item scoring matrix$(\mathrm{u}, \mathrm{i})$, set of nearest neighbour of
   the target users $\mathrm{u}\epsilon\mathrm{u}_\mathrm{n}$ with location, previous N- recommended items
2. Output: Set of items recommended at the location of target user
3. Begin:
4. For each $i_k = \{i_1,\ i_2 \dots i_k\}$
   a. If$i_k$ $is$ not scored by users
      Then scored $i_k$= Avg. score of items

   b. Else {Average all items by mean of their scoring}
         Scored $i_k$ =original score –mean score
5. Convert the user-scoring matrix into user-item preference matrix using Eq.(2.1).
6. Use Pearson coefficient to calculate correlation between users for the similar items
   using Eq.(2.2)
7. Calculate intermediate preference score for the recommended item for the target
   user using Eq. (2.3) and Eq.(2.4)
8. Calculate the asymmetric preference and time weight decay of the item using Eq.
   (2.5) and Eq. (2.6) respectively.
9. Predict the preference score for all the items for the target users using Eq.(2.7)
10. Recommend Top-N list of items to the target user
11. End
```

The proposed collaborative filtering algorithm (CFA) is presented in algorithm 1. Initially, data structure is defined for the users $u_p = \{u_1,\ u_2, \dots u_i \dots . u_p\}$ and for the items $i_k = \{i_1,\ i_2 \dots i_k\}$ with their location and previously top N-recommended items for the target user is feed as input to the algorithm. Line 4 checks the whether the item is previously scored by the user or not, if it is not scored then scored with average value of the items. Thereafter, the line 5 to line 9 involved in scoring the preferred item using correlation, asymmetric influence and time weight decay function of the items. Finally, line number 10 output the top N-recommended items for the target users. The complexity of the proposed CFA algorithm is depending upon Line 4 i.e., number of items checked in for loop and line number 5 matrix multiplication transformation for user-items preference matrix using Eq. (2.2). Further line 6 to line 10 requires constant time to evaluate the parameters. Thus, the total time complexity of the CFA algorithm is $O(k + k^3)$, where k is number of items.

2.5 Simulation and Results

In this section, for the simulation purpose structure model of IoT platform is setup by the sensors, for the data acquisition energy is supplied to sensors continuously. The dataset is used in the simulation are Breast_Cancer_dataset, where each sample contain nine attributes as scoring level. The data size is big, so we adjusted original data set and extracted some samples as items for the experiment, including 72235 rating for 6000 items as sub-data set size of 1GB for different time-intervals. For the testing purpose 80% of extracted data is used for training the model and the remaining 20% data set is reserved for testing the model. The simulation is carried out on the MATLAB 2017b platform.

The evaluation index for the experiments is average recall rate, average accuracy rate and running time on different data volume for the recommended system models. The average recall rate defines the probability that how much user's interested item labelled as favorite item is prescribed by the system. It is the proportion of user's interested item with regard to the prescribed list to the all the items user's resembles in the system as follow:

$$\text{Avg. Rec. Rate} = \frac{\sum_{u\epsilon\ _{tl}} |i_{tl}(u)|}{\sum_{u\epsilon u_p} |i_{tl}(u) \cup i_{rl}(i)|} \tag{2.8}$$

The average accuracy rate is also the probability that how much system correctly predict the user's preference for the items. It is the proportion of the quantity of effectively prescribed items to the absolute number of items suggested by the system as follow:

$$\text{Avg. Acc. Rate} = \frac{\sum_{u\epsilon u_{tl}} |i_{rl}(i)|}{\sum_{u\epsilon u_{tl}} |i_{tl}(u) \cup i_{rl}(i)|} \tag{2.9}$$

Where, u_{tl} represent the all the users in the test list, $i_{tl}(u)$ refers to corresponding item recommended by the user in the test list and $i_{rl}(i)$ represent the item prescribed by the user reside in the recommended list.

2.5.1 Average accuracy vs. data sample set

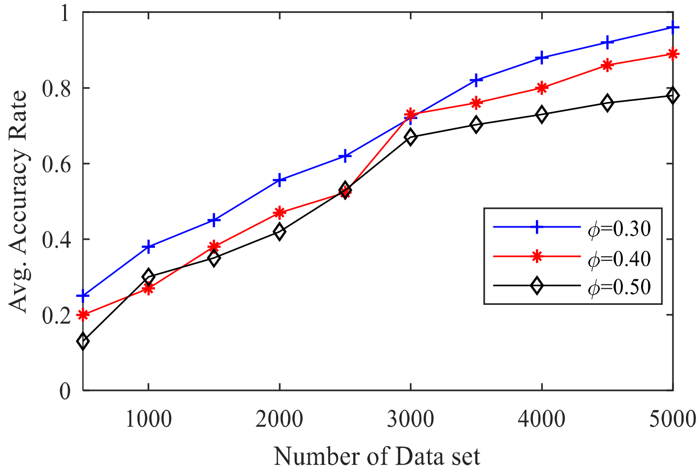


Fig.2.2 Avg. Accuracy Rate over Data set size

Fig.2.2 exhibits that as the number of items or data set is increases, the accuracy of the proposed CFA algorithm is improved. For the simulation purpose we use the time weight decay value from the set {0.50, 0.40, 0.30}. It is also important to note down that as the time weight decay function is decreases the value of accuracy is also increased. This is the because of as the time weight decay function decrease that means the weight of scoring to the item increases, overall average accuracy for the recommended items to the target user is improved. Further, on reaching the data set up to 5000 items, the accuracy of the proposed model reaches almost to 95%. Thus, overall using user-equipment preference scoring matrix and asymmetric influence between users help the proposed model to improve the accuracy of the system.

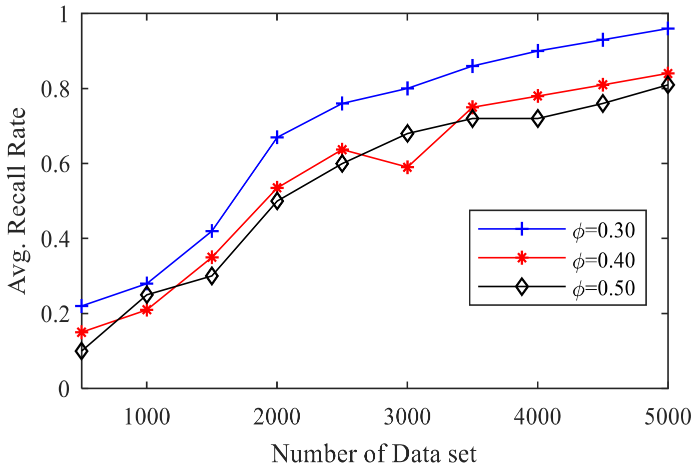


Fig.2.3. Avg. Recall Rate over Data size set

2.5.2 Average recall rate vs. data sample set

Fig 2.3 shows that average recall rate of the proposed algorithm with respect to increasing number of data sample set. For the experimental result we set the time weight decay value {0.50, 0.40, 0.30}. It can be observed from the results as the number of data sample size increase, there is improvement in average recall rate. This is because of initially for the small set of data (items) set, the proposed model is in the learning phase and as the number of data sample size set increased, the average recall rate is increased. It is also attributing to reason the recommended list of the preference item set is also improved as the sample item set is increased. Thus, overall probability of the average recall rate of the favorite item by the target user reaches up to 92% for the proposed model over setting up time weight function 0.30.

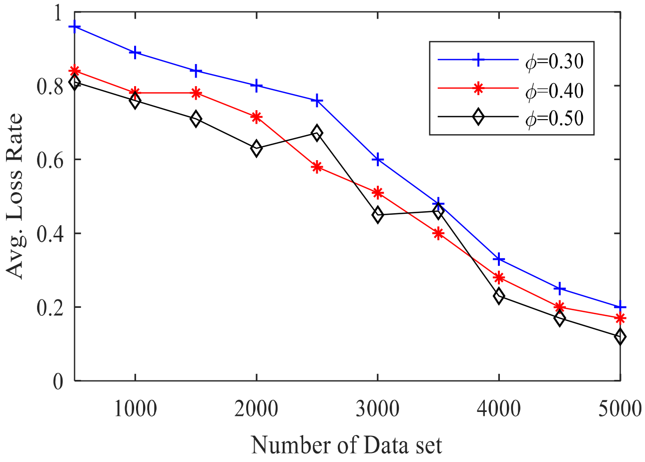


Fig.2.4. Avg. Loss rate over Data set size

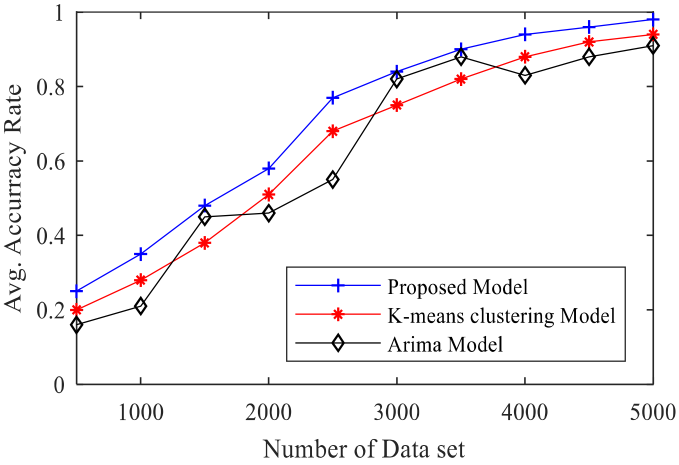


Fig. 2.5. Comparision of Accuracy rate over Data set size

2.5.3 Loss rate vs data sample rate

Fig 2.4. demonstrate the loss rate of the proposed model under different time weight decay value $\{0.50, 0.40, 0.30\}$ for the increasing number of the data set sample value. It can be observed from the results that for the small set of data set data loss rate is about 95% and as there is increase in the data sample set size, the loss rate of the proposed tend to decreases and settled down on 4000 data set size. This is because of as the model extract the more number of items in the recommended list from the values as dictated by symmetrical function influence factor and user-item preference degree, the chance of loss rate is decreases i.e., accuracy rate of the proposed model increases in the sense of recommending the correct items to the target users.

2.5.4 Comparison of Accuracy rate vs state-of-art algorithms

To compare the accuracy rate of the proposed model with state-of-art-models, we set the time decay value is 0.25. This is due to the fact that performance (accuracy rate) of the proposed model is superior when the value of time decay weight function approaches to 0.25as shown in the simulated results of the fig 2.5.As shown in the fig 2.5, with the increment of the data set size, the accuracy of rate of all the models increases linearly. When the data set size is less than 2000, the K-means clustering based model do not perform better than proposed model and the worst performance shown by the ARIMA based model as it shows various up and down throughout on the increasing on data sample set size. This is because of proposed model efficiently take both the influence degree between users for the preference on the similar items and time decay weight value of the items, whereas k-means clustering based model does not count time decay function thus recommendation of preferred item is not accurate. Thus, it is concluding from the results collaborative filtering approach based proposed model outperform than other state-of-art models in terms of accuracy rate for the recommendation of top N-items in the IoT-cloud computing network.

2.6 Summary

This chapter investigate and learned to recommend the top N-list of items to the target user by using data mining approach in the multi-source heterogeneous data available in the IoT-cloud computing network. In this chapter, we consider mainly two aspects. One is to propose system architecture of the IoT-cloud computing that how data are flow from terminal users to application layer. In this regard a four-layer system model is proposed for the data acquisition and mining purpose. On the other hand, a collaborative filtering algorithm based model is proposed to recommend the top rated items to the target user by using time weight decay function and asymmetrical influence degree between users for the similar items. Extensive simulations were performed to show the remarkable performance achieved by the proposed model under different time decay weight value and data sample set size. The simulation results of the proposed model show that accuracy rate; recall rate is about 95% and 92 % respectively on the data sample set size of 5000. In the future work, we consider the energy analysis and their application in different machine learning technique in the IoT-cloud computing platform.

Chapter 3

Q-Learning Enabled Energy Efficient Data Transmission in IoT

This chapter presents a reinforcement based Q-learning energy balanced routing to alleviate the problem of stochastic link quality as channel gain. Using above mentioned parameter an optimization problem has been formulated termed as reward or utility of network. Further, formulated optimization problem converted into Markov decision problem (MDP) and their state, value, action and reward function are described. Finally, a QRL algorithm is presented and their time complexity is analyses. To show the effectiveness of proposed QRL algorithm extensive simulation is performed in terms of convergence property, energy consumption, residual energy and reward with respect to state-of-art-algorithms.

3.1. Introduction

The Internet of Things (IoT) is an internetworking of physical devices, automobiles, houses, as well as as well as sensors, other items embedded with electronics, software, and wireless network connectivity—that collect and exchange data. [93-95]. Each of these smart devices is uniquely identified by Internet address protocol (IP) to forward the data packet from source to destination [95]. The IoT has huge application in almost all the sectors of human being such as healthcare facilities, industrial organization, vehicular network, military operations, business organization and many more [96-97]. These smart devices have limited battery power to perform complex computation and forward the data packet. Due to tremendous upsurge in the connected number of ubiquitous devices, there is large number of data packets travelled in the IoT network. So, there is need to choose the energy balanced routing path to forward the data packet so that lifetime of the network is improved.

In order to support various complicated application, IoT nodes have to perform the reliable operation with their limited energy, computational resources and bandwidth effectively so that it reduce the time-delay for data transmission using shortest routing path, transmission errors and ultimately improve the lifetime of the network. In this regard software define network is combined with the IoT network that separate the hardware and software control operation efficiently to cope with the mentioned challenged [98]. Therefore dynamic routing rules for the IoT nodes provide novel data forwarding strategy, but lack in the presence of stochastic nature of channel path.

Machine learning (ML) techniques have been extensively used in the IoT network to finding the optimal route for data forwarding in the recent decade [99-63]. Machine learning techniques provides learning ability to IoT network through experience, and reinforcement learning (RL) works on learning agent that improve its learning capability based on received rewards according to their taken action. By exploitation of their gained knowledge and exploration of the environment RL agent maximize its rewards [101]. Reinforcement learning techniques requires low computation resources with lower implementation efforts to output effective results with higher accuracy. The output of the system nearly optimal and has higher flexibility according to the changes in the environment without prior knowledge of the network. Thus, reinforcement learning and Q-learning are best suited techniques for routing approaches in the IoT network that build path with lower redundancy.

In [102], authors have proposed Q-learning based algorithm QELAR for the selection of next hop in the routing path. The selection of next hop depends upon the residual energy and the node density of the adjacent node, so that lifetime of the IoT network is improved by evenly distribution of the energy. In [103], authors have proposed multi-sink path selection for the data transmission using the local information such as residual energy, physical distance to update the Q-function. In [104], authors have proposed delay-aware routing algorithm for the underwater sensor networks using

Q-learning. The selection of the next hop is greedy one in the residual energy and minimum propagation delay evaluated through physical distance. Whereas in [105], source nodes broadcast the topology information in the network, then each node simulate the residual energy, distance between them and to the destination node and feed the information to evaluating the reward of the Q-learning function. Then, it creates a virtual topology route for the data transmission and finally data are sent from intermediate node to destination node. However, proposed algorithms have limited computation for constant hop length and fail in the stochastic nature of channel state information. Also, edge length of shortest path routing in the terms of graph is dynamic, which is taken as constant in the above proposed algorithms that are not in the case of real environment.

Under these circumstances, there is need energy balanced routing algorithm based on reinforcement learning approach that include residual energy, physical distance and link quality of the channel for data transmission. The major contribution of the chapter as follow:

1) Firstly, system models consist of network setting, energy consumption with residual energy model and energy balanced routing problem in the IoT network is presented to bring out their primary functions.
2) Secondly, an optimization problem is modelled according to Q-learning and Q-RL based energy balanced routing algorithm is presented. Further, time complexity of the presented algorithm is analysed.
3) Finally, Extensive simulations are presented to check the effectiveness of the presented algorithm in terms of convergence rate, energy consumption, edge length and residual energy with respect to state-of-art-algorithms.

3.2. System Model

3.2.1. Network Setting

We consider an energy-constrained Internet of Thing network that has finite number of sensor nodes, which are randomly deployed in a given monitoring area.Each node in the network can only communicate with the neighbouring nodes that are within its transmission range. Data transmission from one node to another takes place in synchronised time slots. Here, it is considered that each data transmission from a source node to destination takes place by using a number of intermediate nodes present along the route in the network. Each node has a single antenna, a finite battery which can be recharged periodically and works in a half-duplex mode.

The wireless connection between nodes of IoTare affected by many factors, such as residual energy of node, physical distance, channel gain etc. that makes the edge length and network state of dynamic nature in many scenarios. Here, we represent the network as a graphG = (V, E, L)with stochastic edge length, where V is the set of vertices i.e. the sensor nodes and E = (eij), such that vi, vj ϵ V, is the set of edges and L representsthe probability distribution of each edge length. An edge exists between vertices vi and vj in the graph only when node j is the neighbor of node i. The nodes in the transmission range of a node constitute its neighbourhood. The length of edge (vi, vj) is denoted as l(vi, vj) and is considered as a random variable.The channel between neighbouring nodes is assumed to follow quasi-static block Rayleigh fading model and the channel gain$G_{i,j}$ between neighbouring nodes vi andvj are modelled as Markov chain. The transition probability of $G_{i,j}$from $G_1 \; to \; G_2$at any time instant t is given as $\text{þ}_{1,2}^{i,j} = prob\ (G_{i,j}^{t} = G_2 | G_{i,j}^{t-1} = G_1\ )$ and is unknown to the network or sensors.

3.2.2. Energy Consumption and residual energy Model

Energy is consumed in an IoT network for sensing, processing, and communication (transmitting/receiving) activities. Because data communication consumes the majority of a node's energy, only energy consumed for communication is considered during routing. For simplicity, only the energy consumed for transmissions is accounted and energy spent for receiving is ignored as the idle and receiving nodes consume almost same amount of energy [62].According to the first order radio model presented in [62], for a message having bbits, the energy consumed for its transmission from v_i to v_j nodewith edge length l (v_i, v_j) is calculated as

$$E(b,l) = E_T(l) + E_{T_{amp}}(b,l) \tag{3.1}$$

$$E(b,l) = \begin{cases} b*E_T + b*\varepsilon_f * l^2 & if\ \ l < l_0 \\ b*E_T + b*\varepsilon_{amp} * l^4 & if\ \ l \geq l_0 \end{cases} \tag{3.2}$$

After data transmission residual energy$e_h^{res}(t)$ of a node at any hop, at time slot t can be evaluated as follow

$$e_h^{res}(t) \ = \min\{B^{max}, e_h^{res}(t-1) - E(b,l)(t)\ \} \tag{3.3}$$

WhereB^{max} is the maximum battery capacity of a node,$l_0 = \sqrt{\frac{\varepsilon_f}{\varepsilon_{mp}}}$ is used to calculate the threshold distance (l_0) which would be used to calculate the power loss model that could be used, i.e. if either to employed the free space model or fading model. When the spacing with both sender and receiver is below the threshold distance, the free space model is used; otherwise, the multipath fading model is used to calculate the power consumption for transmission purposes.. E_T is the energy requirements of transmitter and receiver circuit, $\varepsilon_f$and$\varepsilon_{amp}$are the energy consumed for amplifying transmission in order to attain a satisfactory signal to noise ratio (*SNR*) and *l*is the communication edge length.

3.2.3. Energy Balanced Routing Problem in IoT

The sensor nodes in the IoT network collect the required data and send it to the destination node. Because of resource limitations in WSNs such as short radio range, limited processing ability, and limited battery power, the source node communicates indirectly through its neighbors (multi-hop) rather than directly with the destination, which results in higher energy efficiency than direct communication. A routing algorithm is required in a multi-hop communication environment to find a communication path from the source node to the destination node. Multi-hop communication eliminates the issue of energy utilization and short-range communication encountered in direct communication, but it causes unequal energy consumption in the network because intermediate nodes lose their batteries faster while relaying data from other nodes. The nodes closest to the sink are the most affected. As a result, the routing algorithm should find a route that equalises the energy consumption of the network's nodes so that all nodes lose their energy nearly at the same time, resulting in increased network lifetime. A routing path rp in the network graph is defined as a sequence of distinct sensors in the WSN starting from source node and ending at destination node i.e. $rp = (v_1, v_2, \ldots\ldots, v_n)$such that $v_i\ and\ v_{i+1}$are adjacent vertices for $1 \leq i\ < n$andv_1=source node andv_n=destination node .The path rphaving n sensor nodes has a length of $n - 1$.The main aim of this chapteris to find an optimal routing path between a source and destination node in order to minimize the total energy consumption and transmission delay for a reliable communication.

3.3. Q-Learning Based Routing Protocol in IoT

In this section, we propose a Q-Learning based efficient routing protocol to find an optimal and reliable route from a source to destination node in order to reduce the total energy consumption and minimize the total transmission delay (based on shortest distance), which can ultimately improve the network lifetime.

3.3.1. Problem Modelling

The stochastic optimal routing-path finding problem is modelled as an MDP. Q-learning updating rules are used to learn an optimal policy. Here, the learning agent selects an action in order to interact with the environment (stochastic graph) to reach the next neighboring node in the route, to get an optimal path from source to destination subject to maximize the expected reward obtained. MDP can be defined as follow:

- State 'S':Each sensor node in the network and the corresponding channel gain towards its neighbor nodes is modelled as a state. The current sensor node in the routing path-finding process and current channel gain is considered as a current state.
- Action 'A':All the out link neighbor nodes are considered in the action set of a state.
- Transition 'P': The next state is determined by theaction selection in current state.

- Reward‘R’: Reward for a state-action pair (s,a) is calculated by using utility value which is the combination of nodes’ residual energy, edge length and nodes’ energy consumption and link quality.

***Definition 1**- (edge length: distance between nodes):* Following formula is used to compute the edge length between any two node v_i, v_j.

$$\mathrm{l}(v_i,\ v_j)=\sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \tag{3.4}$$

where (x_i, y_i) and (x_j, y_j) are coordinates of node $v_i,$ and$v_j$respectively.

***Definition 2-** (edge length based path):* Data packet is transferred from a source node to destination through n-hops towards the destination node. The optimal routing path based on edge length is represented as

$$L_h = \min(l_h, \ldots + l_{h+n-1}) \tag{3.5}$$

Where l_h is the edge length between two nodes at h^{th}-hop. And, the path with minimum edge length guarantees that the transmission delay is minimum.

Definition 2-** (routing path based on energy)**: Data packet is transferred from a source node to destination through n-hops towards the destination node. The optimal routing path based on residual energy is represented as

$$E_h^{res} = \max(e_h^{res}, \ldots + e_{h+n-1}^{res}) \tag{3.6}$$

Where e_h^{res} is the residual energy of transmittingnode ath^{th}-hop. Residual energy of a node can be computed using Eq. (3.3).

Definition 3-** (routing path based on link quality)**: Data packet is transferred from a source node to destination through n-hops towards the destination node. The optimal routing path based on better link quality is represented as

$$L\_Q_h = \max(l\_q_h, \ldots + l\_q_{h+n-1}) \tag{3.7}$$

Where l_q_h is the normalizedlink quality at h^{th}-hopand given as

$$l\_q_h = G_{i,j}^t . \frac{St}{St_{max}} \tag{3.8}$$

Where $St \ and \ St_{max}$ are signal strength at h^{th}-hop and maximum signal strength.

Thus, the reward (utility) obtainedfor transition from $s^t(v_i^t,\ G_{i,j}^t)$ to $s^{t+1}\,(v_i^{t+1},$ $G_{i,j}^{t+1})$ state after taking an action a^tat time slot t being at h^{th}-hop can be computed as

$$R_h^t = W_1 . e_h^{res} + \ W_2 . l\_q_h - W_3 . l_h - \ W_{4.} E(b,l)_h \tag{3.9}$$

Where W_1, W_2, W_3, and W_4 are some prescribed positive weights($\epsilon[0,1]$) parameter which reflect the importance of residual energy, link quality, edge length and energy consumption respectively in calculation of reward, and $j = 1,2, \ldots J$ are number of neighbors for v_i^t. The weight parameters$W_1 \ and \ W_3$are closely related to each other, such that if W_3 is set to zero, the presented model emphasize on the maximization of the residual energy in the routing path and transmission delay is ignored i.e., independent of number of intermediate hop. When $W_1$is set to zero, the presented algorithm pays more attention in reducing the transmission delay of packet and ignores residual energy of the sensor node. Thus, in both above case lifetime of the network is not optimal because of tradeoff between$W_1 \ and \ W_3$. Thus, we can adjust these parameter values according to our needs.

To find an optimal routing path, the learning agent initially perceive the starting state , i.e., the source node and channel gain of the links towards its neighbors, and then selects an action using the current policy, till the agent arrives at the destination node.

The state-value is updated using the temporal difference method. The updating rule for Q-learning is

$$Q(s^t, a^t) = Q(s^t, a^t) + \zeta(R_h^t + \gamma \underbrace{max}_{a} Q(s^{t+1}, a^{t+1}) - Q(s^t, a^t)) \qquad (3.10)$$

where, ζ ϵ [0,1] denotes learning rate, γ ϵ [0,1]is discount factor. Q-learning adopts ∈-greedy policy for action selection, i.e, it select optimal action having maximum state-value with probability 1-∈ and a random action with probability ∈. The main aim of the learning agent is to find an optimal policy $\pi(s^t)$ for selecting an optimal routing path. The optimal policy $\pi^*(s^t)$ denotes the state-value which is greater than other policy's state-value.

The optimal routing problem can be expressed as

$$rp = \underset{\{v_1,v_2,\ldots v_n\}}{argmax} \; lim \, \mathbb{E}[\textstyle\sum_{h=1}^{n-1} \gamma^h R_h^t] \qquad (3.11)$$

$Subject\ to -$

$$l_h \leq l_h^{max}, \;\; \forall\, h = 1,2, \ldots n-1$$
$$e_h^{res} \geq e_h^{res\,(\min)}, \;\; \forall\, h$$
$$l\_q_h \geq l\_q_h^{max}, \;\; \forall\, h$$

The formulated problem in (3.11) can be optimally solved by a Q-Learning method. The goal of proposed routing problem is to maximize the total reward over all routing paths starting at source node, such that rp ∈ φ andφ is the set of all paths in G starting from source node and ending at destination.

In routing process, each node in the network keeps a routing table which is used to select the next hop for the data transmission. The routing table contains the information about next possible nodes which can be reachable to all possible destinations in the network. This table is updated after each data transmission to store the information of node which is good for further data forwarding based on the obtained reward.

3.3.2. Q-Learning based Routing Algorithm

In this section, we present a learning based routing algorithm, particularly using Q-learning approach.

Algorithm- QLRA

```
Q-table Initialization
Initialization of path set ᵽ
D = $v_n$          # destination node
Episode=0
For Episode ≤ $\text{Episode}_{max}$
$s^t = v_i^t,\ G_{i,j}^t$# current state at time slot t
Sr= $v_i^t$              # source node
    ᵽ = Sr
    While Sr≠ $v_n$
Action_set= neighbor nodes of Sr
        Z = Action_set/ ᵽ
        If Z is empty  then
            Break
        End if
$a^t$ = ∈-greedy(Sr) in Z based for $s^t$ state     # action at $s^t$
        Obtain  $R_h^t$and$s^{t+1}$ after $h^{th}$-hop data transmission at $s^t$ state
            #Q-table Update
        $Q(s^t, a^t) = Q(s^t, a^t) + \zeta(R_h^t + \gamma \underbrace{max}_{a} Q(s^{t+1}, a^{t+1}) - - Q(s^t, a^t))$
$s^t = s^{t+1}$
        ᵽ= ᵽ U Z
    End While
    Episode= Episode+1
End For
Final_route= ᵽ
Return Final_route
```

Initially, we initialize the Q-table with all zero values. After that current state $s^t$is observed. Based on the current state an action a^t is selected from the available actions at s^t (line no. 15). After executing the action, a reward and next state s^{t+1} are obtained (line no. 16). Using the achieved reward, state-value $Q(s^t, a^t)$ is updated (line no. 17). And now next state s^{t+1} becomes current state. The algorithm converge either source node find a routing path to reach destination node or for the maximum number of episode.

Time Complexity:

The time complexity of the Q-learning based routing algorithm mainly has three aspects: (1) the algorithm continues until it find destination node in the line no.9 i.e., number of intermediate node in the routing path is $(n - 1)$. (2) Selecting an action (choose neighbor node) from the set of neighbors' nodes subject to maximize the expected discount reward in the network. The set of neighbors is represented in the form of $n \times n$ matrix i.e., but for the single node (state) linear search apply on the single desired row takes $O(n)$ time in line no.10 to line no.15. Thereafter line no.16 to line no. 19 requires constant time to update the Q-value (3).The algorithm runs in worst case are equal to the number of episode until convergence (line no.5). Thus, overall time complexity of the algorithm is $O\big(Episode_{max}(n - 1)n)\big) = O\big(Episode_{max}(n^2)\big)$.

3.4. Simulation Results and Analysis

In this section, firstly, the convergence performance of proposed Q-Learning routing algorithm is analyzed over learning trails and link quality in terms of steps until convergence and reward (utility) respectively. Secondly, comparative analysis of the proposed algorithm against Random learning algorithm and without (w/o) learning algorithm done with respect four metrics: 1) Reward (Utility) 2) Residual energy 3) Energy consumption 4) Edge length. All these algorithms are simulated using same values of parameters for energy model and network conditions.

3.4.1 Simulation Environment

The simulation is carried out using MATLAB in a $100m \times 100m$ square area having 50 sensor nodes are randomly distributed. The communication range and initial energy of all the sensor nodes set to be20 m and 0.5 J respectively. The maximum bandwidth for each of the communication link in the network is 100Mbps. Without loss of generality, one source node and one destination node is selected randomly for the performance analysis of all the state-of-art-algorithms. The others simulations parameters are shown in Table 3.1.

Table 3.1. Simulation Parameters

Parameter	Value	Parameter	Value
B^{max}	$(0.5, 15)\ dBm$	Initial Energy	$0.5\ J$
ζ	0.7	Γ	0.92
$Episode_{max}$	1000	∈	[0,1]
ε_f	3	ε_{amp}	0.9
St_{max}	50 dBm	l_q_h	[2,12]
W_1	0.7	W_2	0.5
W_3	0.3	W_4	0.4
E_T	0.05	n	50
$G_{i,j}$	$10\ dBm$		

3.4.2. Result Analysis

a. Convergence Performance over Learning Trails

Fig 3.1 illustrates the convergence performance of the proposed QRL- based energy balanced algorithm over number of learning trails. It can be clearly observed from result, the RL agent takes 940 steps to converge for the first trails of learning rate thereafter it took less number of steps approximate 560 steps for the very less value of learning rate$\zeta = 0.005$.The number of steps on average higher than 200 steps for the taking the value of$\zeta = 0.02$, whereas the best performance to achieve convergence by the RL agent for the value of$\zeta = 0.05$. Thus, it is necessary to choose the learning rate with caution for convergence in small number of steps towards maximization of reward i.e. reduce the energy consumption and minimize the number of hop counts.

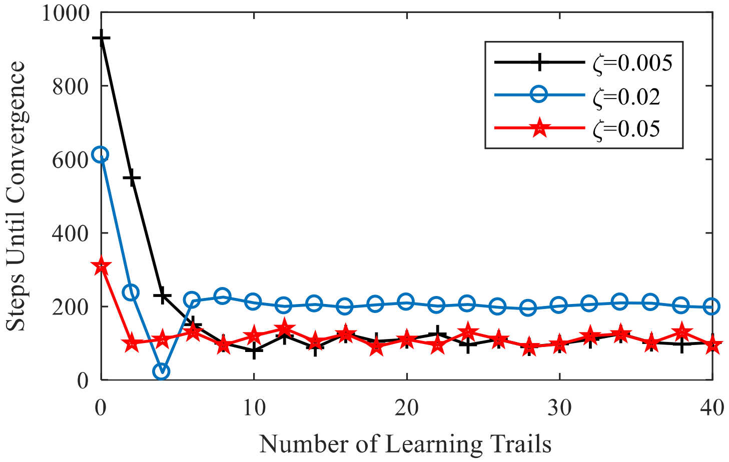


Fig.3.1. Convergence performance over learning trails

b. Reward (Utility) over Link Quality

c.

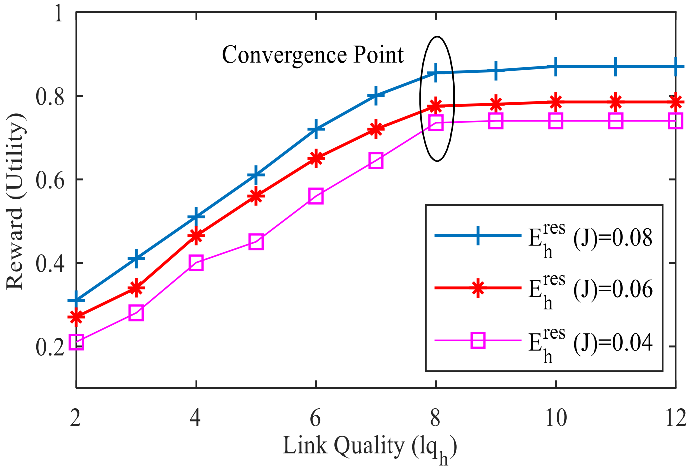


Fig.3.2. Reward (Utility) over link Quality

Fig 3.2. illustrates that the reward (utility) of the proposed Q-learning routing algorithm with respect to link quality under different residual energy of the intermediate hops in the routing path. The link quality describes the nature of communication path between the sensor nodes. And further, link quality depends upon the residual energy of the intermediate hops. If the residual energy of the communicating nodes is high then the link quality is also better and vice-versa. It can be observed from the results, at the beginning, the reward of the proposed algorithm increases rapidly then become stationary as the value of link quality improves. This is because of the learning capability of the proposed algorithm to optimize the reward faster at link quality's value 8. It is also worth to note down reward of the proposed algorithm cannot increase with further increase in the link quality. This is due to the fact other factors such as limited bandwidth and energy consumption in communication also increases and then affects the reward of the proposed algorithm according to Eq. (3.10).

d. Comparative analysis of Reward (Utility) over Episodes

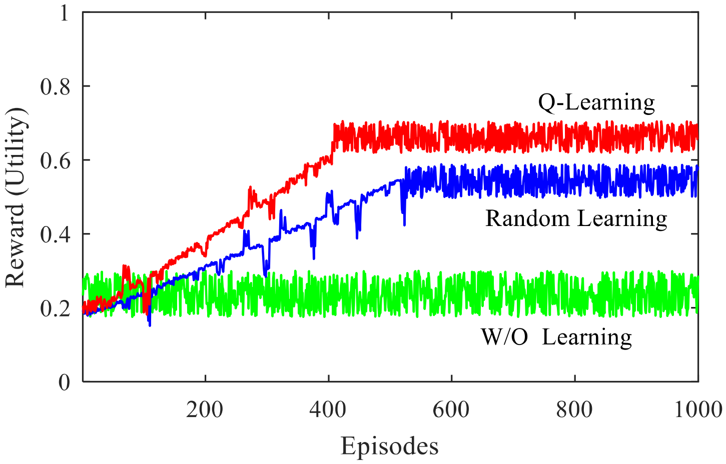


Fig.3.3. Reward (Utility) over Episodes

A comparison of reward (utility) between proposed QRL- based energy balanced algorithm and state-of-art-algorithm over number of episode is presented in the fig 3.3 using learning rate of$\zeta = 0.05$. it is clearly observed from the simulation results that Q-learning algorithm converges faster than random learning algorithm within 390 episodes. Whereas random algorithm's utility converges around 580 episodes. This is due to the fact the proposed QRL algorithm uses ϵ-greedy technique to select an action rather than randomly selected any action for the current state-reward.

Also, optimal learning policy helps in the selection of an action in QRL, which ultimately maximize the reward with less number of episodes. It is also worthy to note down that worst performance is shown by without learning algorithm. This is because of neither have learning policy nor any optimization technique involved in the process of reward maximization.

e. Comparison of Residual Energy over Episodes

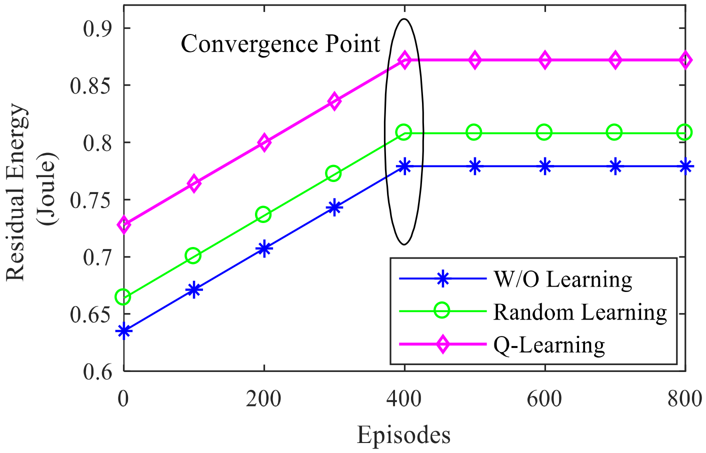


Fig. 3.4. Residual energy over Episodes

A comparison of convergence characteristic in the terms of residual energy between Q-learning and the state-of-art-algorithm is presented in the fig 3.4. using learning rate of$\zeta = 0.05$. It is clearly observed from the result as the number of episode increases residual energy increases for all the three algorithms and converges at 400 episodes. Further, it is noticeable that proposed QRL based energy balanced routing algorithm has higher residual energy about (0.87Joule) than other state-of-art-algorithm. This is because of the. QRL selects the next hop for the routing purpose based on optimal policy learning strategy subject to maximize the residual energy, better link quality and minimum distance. Whereas random algorithm select the next route based on minimum distance and does not consider residual energy of the next hop that in turns increases the overall energy consumption. And, the without learning based algorithm selects any hop for the routing randomly without considering the residual energy and minimum distance.

f. Comparison of Energy Consumption over Episodes

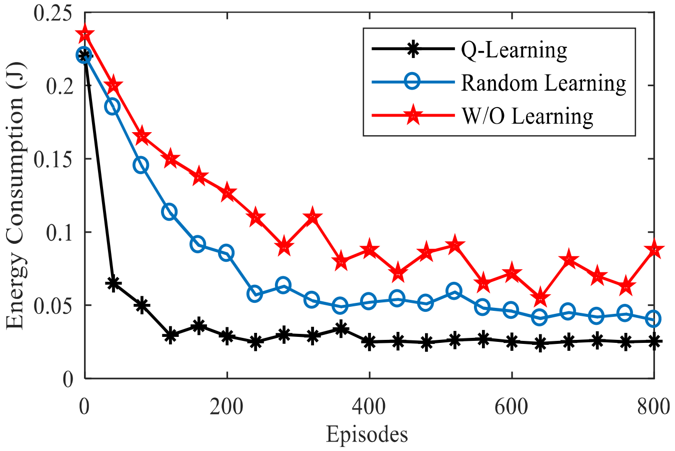


Fig.3.5. Energy Consumption over episodes

A comparison of energy consumption of the proposed Q-learning routing algorithm with state-of-art-algorithms over number of learning episodes is shown in the fig. 3.5. It can be observed from the results; at the start of learning episodes the energy consumption of the proposed algorithm is 0.225 J and as the algorithm reached to 160 episodes it lower down the energy consumption at 0.03 J. Further, proposed algorithm reached up to 400 episodes, the energy reduces to 0.025 J and consumption becomes stable. Whereas other state-of-art algorithm fails to optimize the energy consumption of the nodes involved during routing path. This is because of Q-learning algorithm uses ∈-greedy approach for the selection of optimal policy, whereas Random algorithm selects any action randomly to obtain the reward. It is also noted down that the worst performance is shown by without (w/o) learning algorithm, because it does not have any learning policy and compute reward on the current situation of node's parameters.

g. Comparison of Edge Length over Episodes

A comparison of edge length of the proposed Q-learning routing algorithm with state-of-art-algorithms over number of learning episodes is shown in the fig 3.6. The edge length describe the distance between the intermediate hops, smaller the edge length (Euclidean distance) corresponds to reduce the transmission delay and improves the also convergence speed of the learning based routing algorithm.

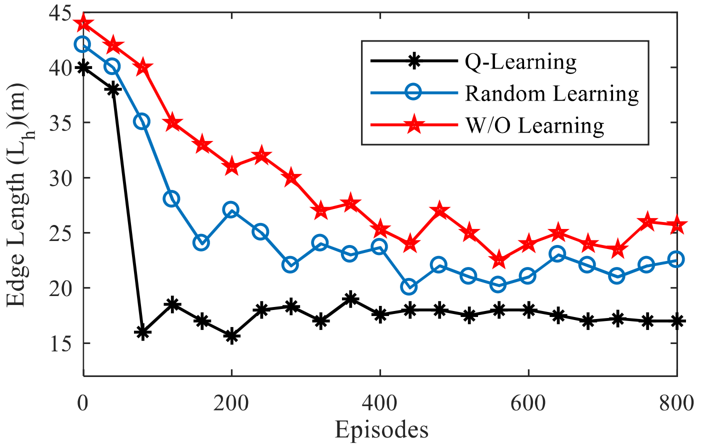


Fig. 3.6. Edge Length over episodes

The results shows that as the number of episode increases, the edge length of proposed algorithm reduces and stabilized about to 17 m within 400 episodes. Whereas edge length of random and without learning algorithms are fluctuates and fails to convergence. This is because of proposed Q-Learning routing algorithm at the initialization know the number of hop counts and also in learning phase ∈-greedy policy helps to compute less number of intermediate hops count and then compute the shortest distance edge length. It can be also observed that without learning based routing algorithm compute the edge length only on Euclidean distance formula according to Eq. (3.4) and nothing to do with learning and in turn fail to converge the edge length.

3.5. Summary

In this chapter, we handled the problem of energy balanced routing algorithm using reinforcement learning and proposed QRL algorithm for wireless sensor network. The link quality, residual energy, and distance between the two consecutive hops are used as parameter for selection of an optimal action subject to maximize the reward (utility). To achieve the objective QRL based energy balanced algorithm has been proposed and their time complexity is also analyzed to show the effectives of the proposed algorithm. It is also proved from the simulation results the proposed QRL algorithm converges faster than other state-of-art-algorithms. It is also notable from the simulation results that energy consumption and link quality and residual energy also improved compared to random algorithm and without learning algorithm. In the future, we also include the node density as another parameter to estimate the energy balanced routing path using deep learning techniques.

Chapter 4

Secure Data Deduplication and Data Sharing in IoT

This chapter presents a fog-enabled secure data deduplication and encrypted search (FDES) scheme for IoT, which uses the Merkle tree for storage of data. The FDES scheme supports dynamic ownership management, and it provides authorized access to the shared data with quantum-resistant attacks using lattice cryptography. We further ensure a multi-user search for the encrypted data using an encrypted keyword based on the homomorphic function. Moreover, each authorized user in the system has his own unique keys which further simplifies key revocation. Security analysis depicts that the performance of FDES scheme is better than the current data deduplication scheme. Moreover, simulation results demonstrate that encryption and decryption time is significantly reduced.

4.1. Introduction

Internet of things (IoT) is a technology, where data sensed by smart devices such as sensors, mobile phones and vehicles, are transmitted to the cloud. Due to limited storage and low computing capability of IoT devices, the network suffers from problems such as the transmission, storage, and security of data. Fog computing reduces transmission delay between user request and cloud response by being closer to the user's data. Fog computing is an emerging network framework with a combination of IoT and cloud computing in which services are handled either at the network level or at the remote data center cloud. To optimize disk space, data deduplication was proposed which stores only one copy of duplicate data and provides an access link to the owners. This saves significant disk space and bandwidth. However, shared usage of data can compromise the security of the network and raises new challenges.

As data owners are worried about unauthorized access of the data from cloud service providers or adversaries, at user level encryption of the data before outsourcing may be considered. To maintain data privacy and deduplication, convergent encryption (CE) [106] is a feasible solution. It computes the hash value of the data file and runs the convergent keys to encrypt and decrypt the sensitive files. After encrypting the file, the data owner outsources the encrypted file to the server and only carries the encryption keys. CE results in the same hash value and the same convergent key. However, CE also witnesses the following problems. First, the revoked users may access the data even when they are no longer authorized user, or leak their shared keys and enable another illegitimate user to access the shared data. Second, efficient key management is another issue in secure data deduplication. Every updated data needs a fresh shared key and thus all the legitimate data owners get the new keys. This can incur significant computation, communication, and storage overhead expenses. Therefore, dynamic ownership and efficient key management are vital for secure deduplication implementation in a fog environment.

Moreover, standard security algorithms such as elliptic curve cryptography, ElGamal, attribute-based encryption will become vulnerable in the era of post-quantum because quantum computers have the capability to solve these algorithms in polynomial time. Hence, advanced security algorithms are necessary to maintain the integrity and confidentiality of fog computing. In this context, lattice-based cryptography is a prominent solution against quantum attacks. The reason for resistance from quantum attacks is that it cannot be decrypted in polynomial time and has been proven secure against various known traditional cyber-attacks [107].

CONTRIBUTION

In this chapter, we propose a secure data deduplication scheme by using Merkle tree with lattice cryptography to save cloud and fog's storage across various service providers and to ensure secure, searchable and dynamic access to the shared data. We designed dual folds' data encryption mechanism particularly at the user level and fog-server level to avoid data leakage at the fog and cloud server levels. Moreover, a multi-user search using encrypted keywords based on the homomorphic function to perform search queries over encrypted data is presented. We also design a mechanism for dynamic updates for all authorized users where any user can insert encrypted data and another user can decrypt it without knowing the keys of others. Security analysis of the FDES scheme has been performed to prove its robustness. In addition, with numerical and experimental results, we reveal that the FDES scheme has improved the computational, communication and storage cost. Finally, we present open issues for fog enabled data deduplication for various IoT environments.

4.2. Related Work

With the rapid development of cloud computing, many deduplication techniques have been proposed. The fundamental solution for the secure deduplication was convergent encryption (CE) [106]. In this scheme, the hash value of the data was used as a key to encrypt the data and it ensures the same ciphertext corresponding to the same data. Further, to guarantee data ownership, proof of ownership (POW) was proposed in [108] to verify the user's data. Then various schemes were put forward for POW using Merkle tree and homomorphic verifiable tags [109]. A novel deduplication scheme [109] was proposed, where the user can generate its file tags. These file tags are used in uploading the file to the cloud server and used for auditing of data while data stored in the cloud server. Further, to prevent data leakage, a server-side deduplication scheme on encrypted data was proposed in [110]. This scheme provides a secure ownership group key distribution and also controls dynamic changes at the cloud server by exploiting randomized CE. A dynamic integrity verification scheme [111] was proposed, which uses a bloom filter and a lattice-based signature. This model uses a vector for confirmation of a collection of elements, which results in better efficiency. However, this model works independently and lacks in providing the user's privacy to improve the verification efficiency of trust providers. For dynamic user management, a secure and scalable data deduplication was proposed in [112]. It dynamically updates the group users and limits the unauthorized cloud users to access the sensitive data owned by valid users. The above-discussed schemes mainly solve the problem of data deduplication and lack of encrypted data searches in a cloud environment.

Searchable encryption (SE) searches keys over encrypted data in the cloud environment, was originally introduced in [113]. The computational overhead in the SE was linear to the size of the search query. Many SE schemes were proposed to resolve this problem. A fully trusted third party [114] was introduced to maintain the secret keys of authorized users. Later, multi-user data sharing scheme [115] was proposed, and it was based on attribute-based SE. The existing SE schemes are based on traditional complexity assumptions and may be compromised by quantum computers. Therefore, it is significant to study how to achieve a secure and efficient data deduplication and encrypted search for post-quantum cloud computing.

4.3. The FDES Scheme

In this section, we propose a secure data deduplication and encrypted data search scheme in the fog environment that has dynamic ownership management capability with quantum-resistant attacks. The FDES scheme is partially based on lattice-based cryptography with Merkle tree storage structure to provide POW. The server stores the data in encrypted form, and homomorphic function helps in search over encrypted data. Fig 4.1 shows the various functions and overview of the FDES scheme.

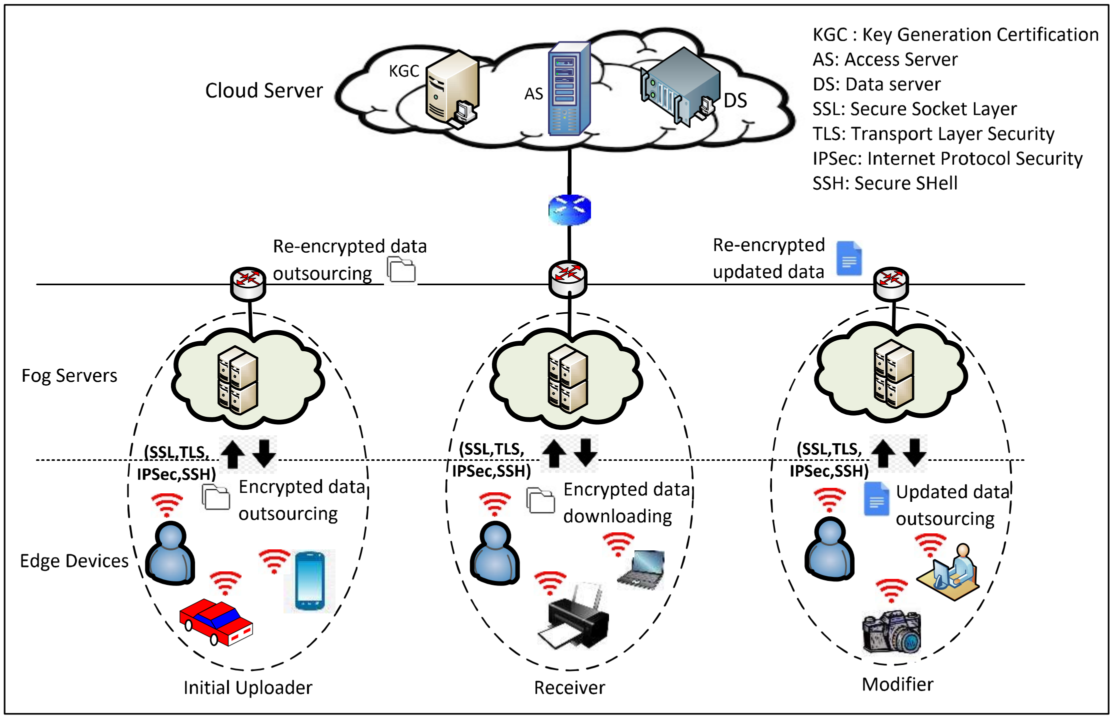


Fig.4.1. Architecture diagram of data deduplication and encrypted search scheme

4.3.1. Scheme Construction

The initial user encrypts the data and then outsources it to the fog server with file-specific keywords. The fog server re-encrypts the data, keys, and keywords. Further, a fog server generates a Merkle tree corresponding to the cipher texts and stores only its root node into its directory and outsources ciphertexts to the cloud. After the initial data upload, the subsequent user attempts to upload the data already stored in the cloud server. Fog-server performs the ownership verification and after successful verification, it adds the subsequent user to the ownership list. To access the file stored in a cloud server, an authorized user sends the request based on an encrypted file-specific keyword search. The fog server re-encrypts the keywords and either search into its own directory or forwards the request to the cloud server. On successful search, the server returns the corresponding encrypted data with the file-specific keys. The scheme supports secured updates over encrypted data. The FDES scheme consists of four phases: {System initialization, Data outsourcing, Fog assisted privacy-preserving data search, and Get updates} and are described in the following subsections.

System Initialization: The key generation certification (KGC) initializes the public parameters pp where the input is lattice polynomial vectors. These parameters are distributed to $fog\ server$ and $users$ of the network. The private key for $fog\ server$ and $user$ are generated from lattice vectors. KGC publishes the pp {user private key, user public key, fog server public key, fog server private key, two hash functions}.

Data Outsourcing Phase:

Step 1. Suppose$uploader$ requests to upload a $file$ to the $fog\ server$with the $hash\ value$and $MerkleRoot$ of the file.

Step 2. The $fog\ server$ first checks the hash of the file in its repository. If it already exists, then deduplication happens at the local repository of$fog\ server$, otherwise it requests a deduplication check to the cloud server. If cloud server finds duplicates, it checks the authenticity by matching the *Merkle Root* stored in its own repository.

Step 3. a) If the *users'* calculated *root value* is equal to the *Merkle Root* stored in the cloud thenserver authenticates it and adds$uploader$ into the ownership list of the *file* at the cloud otherwise, the request gets cancelled. If *hash* of the file does not exist, *uploader* becomes the initial uploader of the file.

Step 3. b) *Fog server* calculates the *file public key* and update the *ownership list.*

Step 4. After receiving the *file public keys*, the user generates public and decryption key corresponding to file.

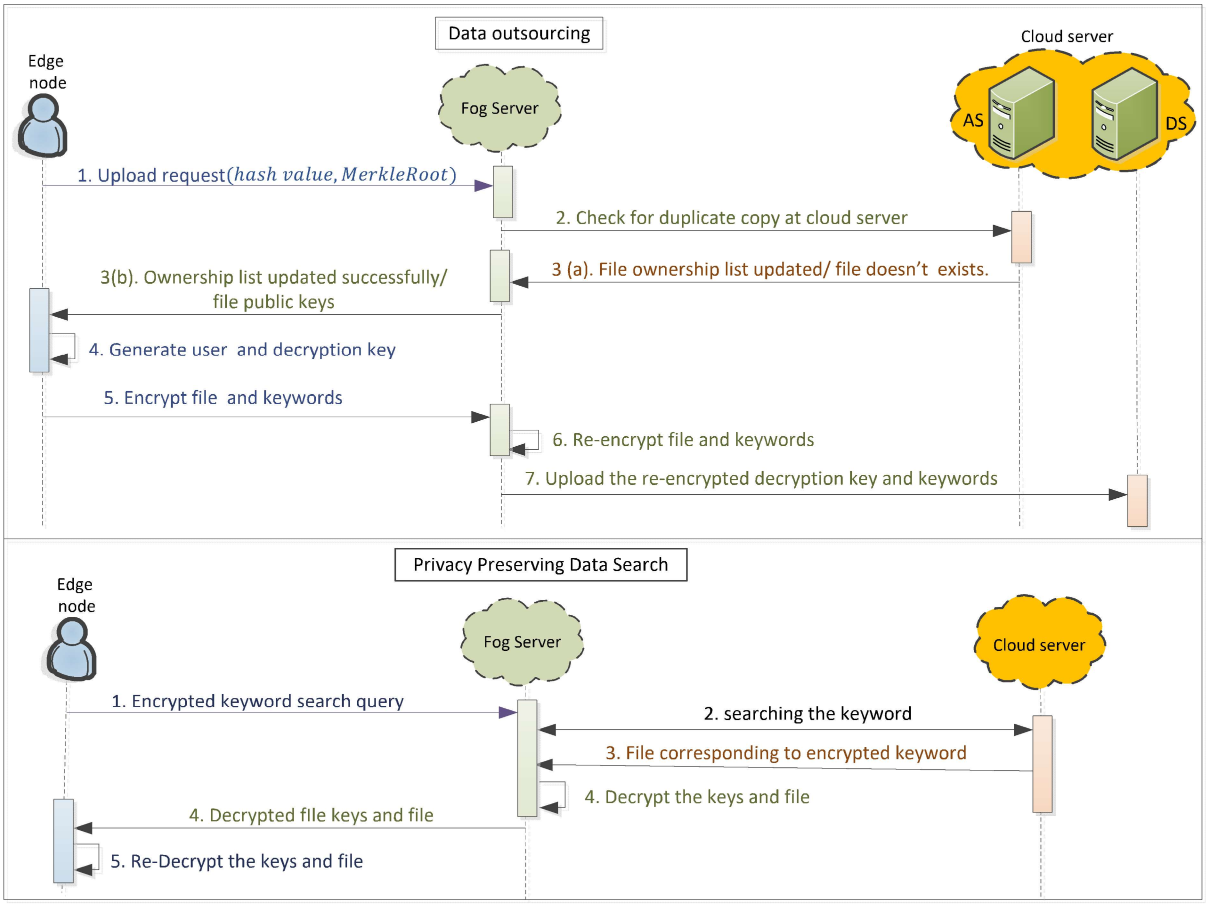

Data outsourcing
Edge node
Fog Server
Cloud server
AS
DS
1. Upload request(hash value, MerkleRoot)
2. Check for duplicate copy at cloud server
3 (a). File ownership list updated/ file doesn't exists.
3(b). Ownership list updated successfully/ file public keys
4. Generate user and decryption key
5. Encrypt file and keywords
6. Re-encrypt file and keywords
7. Upload the re-encrypted decryption key and keywords
Privacy Preserving Data Search
Edge node
Fog Server
Cloud server
1. Encrypted keyword search query
2. searching the keyword
3. File corresponding to encrypted keyword
4. Decrypt the keys and file
4. Decrypted fIle keys and file
5. Re-Decrypt the keys and file

Fig. 4.2. Lattice-based data deduplication and encrypted search

Step 5. *Uploader* encrypts the *file* and search keywords from the standard keyword set corresponding to the *file*. It outsources the *cipher text file* and *cipher-keywords* to the local *fog server*.

Step 6. The outsourced data file is very valuable and unauthorized access is unacceptable for the data owners. To protect the data, the *fog server* re-encrypt the *cipher keywords* and *key* which are uploaded by *uploader*. Further, the *fog server* computes the *hash of cipher blocks* and generates *Merkle tree* as shown in Fig 4.2. Finally, the fog server stores the tuple *(Merkle tree, encrypted key, and encrypted cipher keywords)* into its entry table.

Step 7. *fog server* uploads *re-encrypted cipher file* to the *cloud server* for storage space purpose. The procedure for data outsourcing is depicted in Fig 4.2.

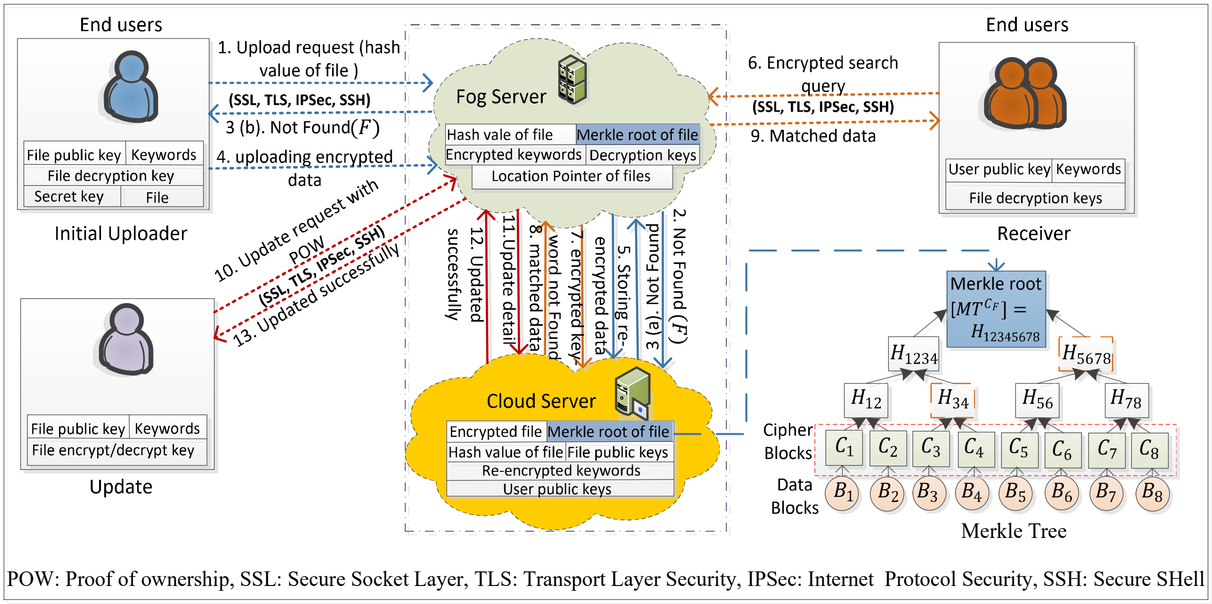
4.3.2. Fog Assisted Privacy-preserving data search

The data search occurs at *fog* and *cloud servers*. Before starting the search, *fog server* encrypts the *cipher keywords* and then searches for the document. The encrypted keyword search helps in maintaining file anonymity. Fig 4.2 illustrates a typical procedure.
Step 1. When a *user* wants to access the files corresponding to a keyword, it encrypts the *keyword* and asks the fog server whether the requested file exists.
Step 2. The *fog server* first checks whether corresponding cipher keywords exist in its *trending keyword list* or not and then checks in its entry table. The trending keyword list helps to reduce searching time. If the fog server finds the file along with keys, it doesn't consult the cloud server, otherwise, *re-encrypt* the keywords and forward the request to the cloud server.
Step 3. The *cloud server* sends the file along with keys to the fog server.
Step 4. The fog server decrypts the keys and encrypted file, and outsources it to the *user.*
Step 5. *Data decryption: User* first re-decrypts the key and then file using the file decryption keys. An unauthorized user cannot decrypt the file because the server does not have the corresponding proxy decryption key. Finally, *user* gets the plain *file.*

Get Updates: Users update their outsourced contents from the remote storage by performing the POW as follows:

- *POW*: A user performs verification before updating the data file to prove the ownership with the help of Merkle tree.
- *Modify:* As we already know, the data stores in encrypted form. Server performs the update operations on cipher text with the help of homomorphic functions. The *updated data* is encrypted with the help of file-specific keys and is outsourced to the nearby fog server along with an updated *internal state*. Fog server inserts the cipher text and updates the corresponding *Merkle tree.* After updating the file, state of the file changes from *internal* to *outward* state and all the users of the file get the information about the update. Authorized *data owner* uses extended Euclidean functionto get the original *plaintext* through the information obtained from the fog server.

- *Delete:* If the *user* wants to delete the *data* from the server, the user requests the fog server with the request ($req = delete$). The fog server checks the requested file in its local repository. If fog server finds the file, it delete the *data* otherwise asks the cloud server to update the file. The cloud server update the files and changes the state of file to *outward* and inform all the owners about the modification.



POW: Proof of ownership, SSL: Secure Socket Layer, TLS: Transport Layer Security, IPSec: Internet Protocol Security, SSH: Secure SHell

Fig. 4.3. Overall flowchart of the fog-assisted deduplication and search over encrypted data

4.3.3. Overall Flowchart of The FDES Scheme Over Encrypted Data

In this section, we present a flowchart of the FDES scheme as shown in Fig 4.3. The IoT devices collect and aggregate the data. Then the data is outsourced to the fog server. After receiving the data, the fog server checks the duplicate copy at the local storage. Further, the fog server asks to remove the redundancy at the cloud storage to optimize the storage space. The fog servers only store the frequently searched files. In addition, to maintain the privacy of users' data, the outsourced data re-encrypted at the server level before being stored it into the database. To improve search efficiency, a user also outsources multiple searching keywords in the encrypted form corresponding to a data file. Further, the fog server stores the keywords after re-encryption and outsources all the data files in the re-encrypted form to the cloud server. When a user needs to access the data, it sends the encrypted keyword query to the local fog server, and it checks its repetition rate. If the repetition rate is greater than 0.6, the fog server holds the file else, transfer the request to the cloud server. The cloud server searches encrypted keywords in its repository and sends the corresponding encrypted file to the fog server. Further, fog server decrypts the file based on respective keys and sends the encrypted text to the data users. The data users decrypt the data with the help of corresponding data keys.

Considering advance IoT environment, data is changing rapidly which creates the need to update the data at the server-level. In order to update the data, the user can modify the uploaded files and delete it as well with the help of fog server.

4.4. Security Analysis

In this section, we conduct a security analysis of the FDES scheme on various attributes and comparatively analyze the scheme with previously related works and are shown in Table 4.1.

Dynamic update: The FDE Sup dates dynamic file and ownership list with the help of re-encryption technique in fog storage. The cloud server maintains changes in the file and ownership list. On getting an update from the fog server, cloud server immediately updates the file at its respective locations.

Quantum computer attack resistant: Sharing sensitive information over the network can cause serious security and privacy concerns because of quantum computers. The FDES scheme is partially based on a lattice-based cryptography with homomorphic functions which renders security against the quantum attacks.

Whether the third party can obtain user files: In the FDES scheme, cloud server maintains an ownership list corresponding to a file. Any user requesting for a file can access the file if they prove the ownership by giving a correct value of *Merkle Root* to fog server. So, it supports multiple users' access.

Privacy-preserving ownership proof: In the FDES scheme, fog-server and data owners prove their ownership via a public channel. Hence, the public parameters (*hash of key, file public keys*) can be obtained by an adversary. Since, the plaintext is encrypted using a tag, where the tag comprises cryptographic hash functions that are pre-image resistant. So, if an adversary gets the cipher text, it cannot recover the plaintext. Furthermore, the file-specific public keys are drawn using lattice vectors, so it reveals no secret key using the combination of the keys.

The secret keys are randomized and can be considered as short-term keys. The fog-server cannot derive the keys with the combination of these keys into its repository. Thus, proof of ownership is secure against any adversary's attack.

Data authentication and integrity check: In FDES scheme, Merkle tree is used to provide data authentication and integrity check. The verification of data integrity is done in the data outsource phase. In this phase, the fog server provides a series of node values on the path from a leaf node to the root node of the tree. The fog-user generates the entire Merkle tree for the attested data while the fog-server stores only the root node value into its repository. Hence, only the authenticated user can generate a valid Merkle tree.

Secure user-level key management: Every data owner selects its private key independently. Each valid user has a decryption key corresponding to the encrypted key based on extended Euclidean function. *Data decryption key* contains a random *seed value* selected from a uniform distribution so that no adversary can acquire any information from a series of user-level public keys in probabilistic polynomial-time. Only a valid user can decrypt the data. Even if an adversary is a valid data owner, it cannot obtain the user-level private key.

Forward secrecy: Whenever a user modifies or delete a data, the central cloud immediately updates the corresponding file and also update the ownership list, if required. The file-specific keys are also get updated. When a user revoked from the ownership list, the cloud server deletes the corresponding user's ID along with the key and again calculates the file-specific keys. Selective user-level updates in the cloud ensure the forward secrecy against unauthorized access by revoked users.

Table.4.1 Comparative analysis on various security parameters

Security Parameter	[109]	[110]	[111]	[108]	[112]	FDES
Dynamic Update	Yes	Yes	Yes	Yes	Yes	Yes
Quantum computer attack resistant	Yes	No	No	No	No	Yes
Whether the third party can obtain user files	No	Yes	No	No	Yes	Yes
Privacy-preserving ownership proof	No	Yes	No	Yes	Yes	Yes
Data authentication and integrity check	Yes	Yes	Yes	Yes	Yes	Yes
Secure user-level key management	Yes	No	Yes	Yes	Yes	Yes
Forward secrecy	Yes	Yes	Yes	Yes	Yes	yes
Encrypted search	No	No	Yes	No	No	Yes

Table.4.2. Comparison of computation, communication and storage cost

	Computation cost		Communication cost		Storage cost		
Scheme	User	Server	User-Server	Server-user	Public parameters	server	user
[110]	0.0624 *s*	0.00092	2560 *bits*	2464 *bits*	256 *bits*	*2048 bits*	768 *bits*
[116]	0.26944	0.32072	4096 *bits*	–	512 *bits*	896 *bits*	896 *bits*
[108]	0.02896	0.11520	1024 *bits*	1664 *bits*	256 *bits*	1024 *bits*	640 *bits*
[112]	0.10128	0.06832	1664 *bits*	1536 *bits*	256 *bits*	1024 *bits*	640 *bits*
FDES	0.0784 *s*	0.38496	1152 *bits*	1024 *bits*	512 *bits*	896 *bits*	512 bits

Encrypted search: All the fog and cloud servers are not fully trustworthy neither for contents nor for the keys. If a document is called by a particular keyword query, then the server gets to know about the content of the file. Hence, to protect the file and keys from unauthorized access, both are encrypted at user and fog-level. Even if an adversary gets the data from the cloud repository, it cannot decrypt the data without the decryption key. Therefore, encrypted search helps in security across multiple devices and moves data securely.

4.5. Simulation Results and Performance Analysis

In this section, we analyzed the FDES scheme with state-of-the-art techniques in terms of computation, communication, and storage overheads. The theoretical comparative analysis of computation cost of the four schemes [110], [116], [108] and [112] are computed on both user and server sides and summarized in Table 4.2.

- In the FDES scheme, the computation cost is evaluated in two phases, first is user phase and another is server. User phase consists of four *exponential* operations and five *hash* operations, where two *exponential* operations are performed for user key generation and another two *exponential* operations with five *hash* operations are performed for encryption and decryption of file and keywords. In receiver phase, two *exponential* operations and three *hash* operations are used, where two *exponential* operations are computed for decryption and three hash operations are performed for re-encryption of file and keywords. Based on the experimental results [119], the average time required for *exponential* operation is ≈ 0.0192 sec and *hash* operation takes ≈ 0.00032 sec, approximately. So, the total computation time at the user's side is $(4\times0.0192 + 5\times0.00032) \approx 0.0784$sec and at server's side is $(2\times0.0192 + 3\times0.00032) \approx 0.38496$sec.
- For communication cost, assume hash identity to be 256 bits, encrypted file 512 bits, key size to be 128 bits. In FDES scheme, initially user communicates the *ciphertext* of *data file*, the *user public key*, *data decryption key*, and *file user key*. In total it involves one encrypted file, one key, and two hash identity in user-server communication. Thus, bits exchanged are $512+128+256+256 \approx 1152$ bits. Further, server-user requires one encrypted file, two keys, and one hash identity for fog-cloud communication to send the

re-encrypted file, re-encrypted tag, re-encrypted *keywords* and *decryption key.* Thus, bits exchanged in server-user are 512 + 128 + 256 + 128 ≈ 1024 bits.

- In the FDES scheme, cloud server stores the cipher text files along with the keys and public parameters. Uploader only stores the file-specific parameters in its repository and outsources all the data to the fog server. Further, fog server uploads the data file to the cloud server and stores only its root node value with cipher keys. Hence, storage requirement for public parameters is 512 bits to store two elements of groups. Server stores one encrypted file, one group element, and key file. So, the total storage cost at server is512+ 256+128 ≈ 896 bits. User stores one group element and *Merkle Root* in its repository. Thus, total storage bits are 256 +256 ≈ 512 bits.

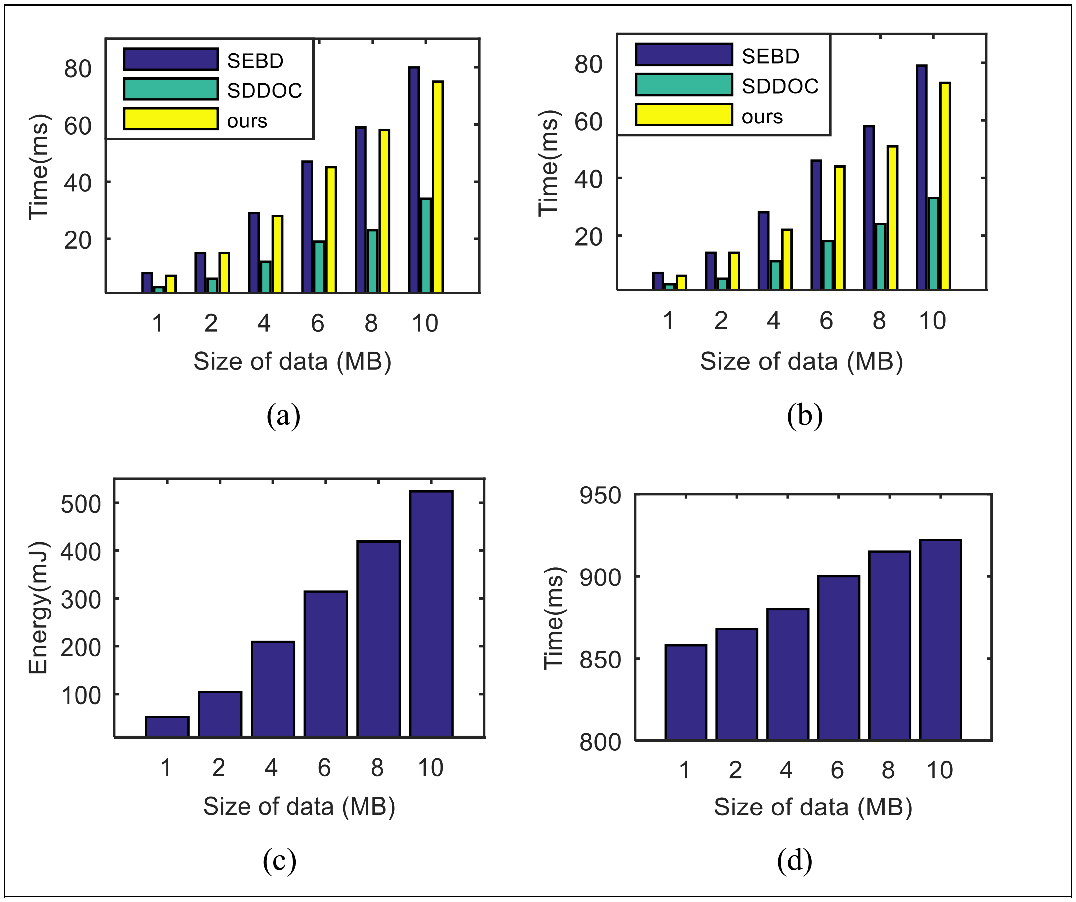


Fig.4.4. Simulation results: a) Computation time for Encryption, b) Computation time for Decryption, c) Total energy consumption vs size of data, d) Total execution time vs size of data.

4.5.1 Software Implementations and performances

To investigate the actual performance of the FDES scheme, we extend the Microsoft lattice cryptography library [117] to implement data deduplication and encrypted search scheme. All the experiments are performed on Microsoft Visual Studio 2013 platform on Intel(R) Core(TM) i7-8700 CPU @3.20 GHZ with a x64-based processor, running Windows 10. For the comparative analysis, we compute the encryption, decryption, computational and total CPU time with varying file sizes from 1 MB to 10 MB. The FDES scheme is compared with SDDOC [108] and SEBD p [110].

In the data outsourcing phase, the encryption time consumed by the FDES scheme is 7ms for 1 MB file while SEDB and SDDOC consume 8ms and 3ms, respectively as shown in Fig 4.4a. This is because of the FDES scheme uses lattice vectors performs better than bilinear pairing schemes [108] and [110]. From Fig 4.4b, it is noticed that decryption time for the FDES scheme, and SEBD is 22ms and 28ms for 4MB file, respectively. From Fig 4.4c, we analyzed that as the data size grows, energy consumption also increases linearly for data deduplication and encrypted search. Fig 4.4d shows the total time taken by the FDES scheme for varying input data size. Time taken by 6MB file is 900ms and it grows further if the file size increases. This is due to the increase in uploading and downloading time with the growth of data.

4.6. Open Issues And Future Directions

We identified various important issues for the fog-assisted data deduplication in IoT networks that require advance and robust solutions. A few such issues are listed as:

Data deduplication for the Internet of vehicle (IoV): For IoV with high mobility, a dynamic approach is required to deal with parking, and traffic lights. Particularly for the cloud server, it will be difficult to maintain data deduplication and reduce the response time in such a high mobility network.

Data deduplication technique for underwater sensor-cloud: In Underwater sensor-cloud, various elements (e.g., ocean waves) might interfere with the transmission process that results in multipath fading. Thus, storing data deduplication for underwater needs to be given emphasis.

Detection of rogue nodes in the fog-assisted network: The corrupted fog nodes may pretend to be legitimate nodes. Such kind of nodes causes serious attacks such as denial of service attacks, signal jamming. Thus, a reliable and robust intrusion detection system for each fog node may reduce the attacks.

Privacy-preserving data aggregation: Data aggregation reduces a significant amount of communication costs. It will be quite significant to develop a multi-user signature key for distinct messages of different types generated by distinct users.

4.7. Summary

In this chapter, a fog-enabled data deduplication and encrypted search scheme is proposed for the post-quantum era. We use the lattice cryptography with homomorphic functions to search over encrypted data which renders the security against the quantum attacks. The search queries are executed by re-encryption of outsourced cipher-keywords at server-level by fog. We conduct the security analysis for data deduplication on various parameters. Further, we compute the computational, communication and storage costs and compare them with state-of-the-art schemes. We perform a comparative analysis of encryption and decryption time. Finally, the FDES scheme is comparatively better than state-of-the-art techniques.

Chapter 5
Conclusion and Future Work

In this thesis, Data mining enabled techniques for IoT have been presented in different ways. The models for Collaborative Filtering based Recommended System, Q-Learning Based data transmission, and data deduplication for cloud and IoT based network have been proposed. The proposed models and algorithms have compared with the existing state of the art algorithms. This chapter presented the conclusion of the work carried out in the thesis, discussion and future direction to extend the proposed work.

5.1 Conclusion

The data around us is useless until it is processed under data mining techniques. With the utilization of Data Mining techniques, the IoT environment becomes intellectual. The work presented in this thesis is a diffident effort to develop data mining assisted model for data transmission for IoT and analyze their performance. Therefore, three different approaches for data transmission have been developed.

We present system architecture of IoT cloud computing platform for data management, storage, analysis and uses data mining approach to recommend top N-priority list to the target users. Collaborative filtering algorithm is presented to improve the prediction accuracy on the basis of user scoring matrix. We also consider the asymmetrical influence degree of the scoring of user on similar items and time decay weight function for the recommended system. In this work, we consider mainly two aspects. One is to propose system architecture of the IoT-cloud computing that how data are flow from terminal users to application layer. In this regard a four-layer system model is proposed for the data acquisition and mining purpose. On the other hand, a collaborative filtering algorithm based model is proposed to recommend the top rated items to the target user by using time weight decay function and asymmetrical influence degree between users for the similar items.

The problem of energy balanced routing algorithm using reinforcement learning has been proposed for sensor enabled IoT network. The link quality, residual energy, and distance between the two consecutive hops are used as parameter for selection of an optimal action subject to maximize the reward (utility). To achieve the objective QRL based energy balanced algorithm has been proposed and their time complexity is also analyzed to show the effectives of the proposed algorithm. It is also proved from the simulation results the proposed QRL algorithm converges faster than other state-of-art-algorithms. It is also notable from the simulation results that energy consumption and link quality and residual energy also improved compared to random algorithm and without learning algorithm. In the future, we also include the node density as another parameter to estimate the energy balanced routing path using deep learning techniques.

A fog-enabled data deduplication and encrypted search scheme is proposed for the post-quantum era. We use the lattice cryptography with homomorphic functions to search over encrypted data which renders the security against the quantum attacks. The search queries are executed by re-encryption of outsourced cipher-keywords at server-level by fog. We conduct the security analysis for data deduplication on various parameters. Further, we compute the computational, communication and storage costs and compare them with state-of-the-art schemes. We perform a comparative analysis of encryption and decryption time. Finally, the FDES scheme is comparatively better than state-of-the-art techniques.

5.2 Discussion

In this work, the results obtained from the proposed data mining enabled models for IoT are presented. Mathematical models and algorithms developed for efficient data transmission in IotT are simulated using MATLAB, for different settings of parameters. After analyzing the results obtained in this work, we observed the following:

- Extensive simulations were performed to show the remarkable performance achieved by the proposed Collaborative filtering algorithm under different time decay weight value and data sample set size. The simulation results of the proposed model show that accuracy rate; recall rate is about 95% and 92 % respectively on the data sample set size of 5000. In the future work, we consider the energy analysis and their application in different machine learning technique in the IoT-cloud computing platform.
- To compare the accuracy rate of the proposed Collaborative filtering algorithm with state-of-art-models, we set the time decay value is 0.25. This is due to the fact that performance (accuracy rate) of the proposed model is superior when the value of time decay weight function approaches to 0.25.
- It can be observed from the results that for the small set of data set data loss rate is about 95% and as there is increase in the data sample set size, the loss rate of the proposed tend to decreases and settled down on 4000 data set size. This is because of as the model extract the more number of items in the recommended list from the values as dictated by symmetrical function influence factor and user-item preference degree, the chance of loss rate is decreases i.e., accuracy rate of the proposed model increases in the sense of recommending the correct items to the target users.
- The convergence performance of the proposed QRL- based energy balanced algorithm over number of learning trails is better as compared with the state of the art algorithms. It can be clearly observed from result, the RL agent takes 940 steps to converge for the first trails of learning rate thereafter it took less number of steps approximate 560 steps for the very less value of learning

rate$\zeta = 0.005$.The number of steps on average higher than 200 steps for the taking the value of $\zeta = 0.02$, whereas the best performance to achieve convergence by the RL agent for the value of$\zeta = 0.05$.

- It can be observed from the results, at the beginning, the reward of the proposed Q-learning algorithm increases rapidly then become stationary as the value of link quality improves. This is because of the learning capability of the proposed algorithm to optimize the reward faster at link quality's value 8.
- It is noticeable that proposed QRL based energy balanced routing algorithm has higher residual energy about (0.87Joule) than other state-of-art-algorithm. This is because of the. QRL selects the next hop for the routing purpose based on optimal policy learning strategy subject to maximize the residual energy, better link quality and minimum distance. Whereas random algorithm select the next route based on minimum distance and does not consider residual energy of the next hop that in turns increases the overall energy consumption. And, the without learning based algorithm selects any hop for the routing randomly without considering the residual energy and minimum distance.
- it is clearly observed from the simulation results that Q-learning algorithm converges faster than random learning algorithm within 390 episodes. Whereas random algorithm's utility converges around 580 episodes. This is due to the fact the proposed QRL algorithm uses ϵ-greedy technique to select an action rather than randomly selected any action for the current state-reward. Also, optimal learning policy helps in the selection of an action in QRL, which ultimately maximize the reward with less number of episodes.
- The results shows that as the number of episode increases, the edge length of proposed algorithm reduces and stabilized about to 17 m within 400 episodes. Whereas edge length of random and without learning algorithms are fluctuates and fails to convergence. This is because of proposed Q-Learning routing algorithm at the initialization know the number of hop counts and also in learning phase ∈-greedy policy helps to compute less number of intermediate hops count and then compute the shortest distance edge length.

- In the data outsourcing phase, the encryption time consumed by the FDES scheme is 7ms for 1 MB file while SEDB and SDDOC consume 8ms and 3ms, respectively.
- The total time taken by the FDES scheme for varying input data size 6MB file is 900ms and it grows further if the file size increases. This is due to the increase in uploading and downloading time with the growth of data.

5.3 Future Work

- In the future, we also include the node density as another parameter to estimate the energy balanced routing path using deep learning techniques.
- As the sensor devices produce a huge amount of data and sensor devices are battery powered, therefore an energy efficient mechanism should be developed with some improvement scopes with the passage of time. Such environment will always be in the scope of improvements. That's why energy issue in IoT environments will always be a hot topic of research.
- Along with energy, the huge amount of produced data may suffer from congestion issues. Therefore congestion management or congestion control could be the sensitivity research topic in IoT environment.
- The sensor devices have a limited storage capacity and this is an era of technology, therefore cloud based or fog based mechanism should be developed with some advanced features to incorporate storage concept.
- For Internet of Vehicles, with high mobility, a dynamic approach is required to deal with parking, and traffic lights. Particularly for the cloud server, it will be difficult to maintain data deduplication and reduce the response time in such a high mobility network.
- Data aggregation reduces a significant amount of communication costs. It will be quite significant to develop a multi-user signature key for distinct messages of different types generated by distinct users.

References

[1]. Atzori, L., Iera, A., & Morabito, G. (2010). The internet of things: A survey. *Computer networks*, *54*(15), 2787-2805.

[2]. Li, S., Da Xu, L., & Zhao, S. (2015). The internet of things: a survey. *Information Systems Frontiers*, *17*(2), 243-259.

[3]. Miorandi, D., Sicari, S., De Pellegrini, F., & Chlamtac, I. (2012). Internet of things: Vision, applications and research challenges. *Ad hoc networks*, *10*(7), 1497-1516.

[4]. Bandyopadhyay, D., & Sen, J. (2011). Internet of things: Applications and challenges in technology and standardization. *Wireless personal communications*, *58*(1), 49-69.

[5]. Domingo, M. C. (2012). An overview of the Internet of Things for people with disabilities. *Journal of Network and Computer Applications*, *35*(2), 584-596.

[6]. Kulkarni, R. V., Förster, A., & Venayagamoorthy, G. K. (2010). Computational intelligence in wireless sensor networks: A survey. *IEEE communications surveys & tutorials*, *13*(1), 68-96.

[7]. Kortuem, G., Kawsar, F., Sundramoorthy, V., & Fitton, D. (2009). Smart objects as building blocks for the internet of things. *IEEE Internet Computing*, *14*(1), 44-51.

[8]. Alam, K. M., Saini, M., & El Saddik, A. (2015). Toward social internet of vehicles: Concept, architecture, and applications. *IEEE access*, *3*, 343-357.

[9]. Alam, M. A. U., Roy, N., Misra, A., & Taylor, J. (2016, June). CACE: Exploiting behavioral interactions for improved activity recognition in multi-inhabitant smart homes. In *2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS)* (pp. 539-548). IEEE.

[10]. Bin, S., Yuan, L., & Xiaoyi, W. (2010, April). Research on data mining models for the internet of things. In *2010 International Conference on Image Analysis and Signal Processing* (pp. 127-132). IEEE.

[11]. Rashidi, P., Cook, D. J., Holder, L. B., & Schmitter-Edgecombe, M. (2010). Discovering activities to recognize and track in a smart environment. *IEEE transactions on knowledge and data engineering*, *23*(4), 527-539.

[12]. Phyu, T. N. (2009, March). Survey of classification techniques in data mining. In *Proceedings of the International MultiConference of Engineers and Computer Scientists* (Vol. 1, No. 5).

[13]. Roddick, J. F., & Spiliopoulou, M. (2002). A survey of temporal knowledge discovery paradigms and methods. *IEEE Transactions on Knowledge and data engineering*, *14*(4), 750-767.

[14]. Chen, F., Deng, P., Wan, J., Zhang, D., Vasilakos, A. V., & Rong, X. (2015). Data mining for the internet of things: literature review and challenges. *International Journal of Distributed Sensor Networks*, *11*(8), 431047.

[15]. Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, *31*(3), 264-323.

[16]. Xu, R., & Wunsch, D. (2005). Survey of clustering algorithms. *IEEE Transactions on neural networks*, *16*(3), 645-678.

[17]. [Ng, R. T., & Han, J. (2002). CLARANS: A method for clustering objects for spatial data mining. *IEEE transactions on knowledge and data engineering*, *14*(5), 1003-1016.

[18]. Guha, S., Meyerson, A., Mishra, N., Motwani, R., & O'Callaghan, L. (2003). Clustering data streams: Theory and practice. *IEEE transactions on knowledge and data engineering*, *15*(3), 515-528.

[19]. Chiang, M. C., Tsai, C. W., & Yang, C. S. (2011). A time-efficient pattern reduction algorithm for k-means clustering. *Information Sciences*, *181*(4), 716-731.

[20]. Anvari-Moghaddam, A., Monsef, H., & Rahimi-Kian, A. (2014). Optimal smart home energy management considering energy saving and a comfortable lifestyle. *IEEE Transactions on Smart Grid*, *6*(1), 324-332.

[21]. Bijarbooneh, F. H., Du, W., Ngai, E. C. H., Fu, X., & Liu, J. (2015). Cloud-assisted data fusion and sensor selection for internet of things. *IEEE Internet of Things Journal*, *3*(3), 257-268.

[22]. Biswas, S., & Misra, S. (2015, November). Designing of a prototype of e-health monitoring system. In *2015 IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)* (pp. 267-272). IEEE.

[23]. Brdiczka, O., Reignier, P., & Crowley, J. L. (2007, September). Detecting individual activities from video in a smart home. In *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems* (pp. 363-370). Springer, Berlin, Heidelberg.

[24]. Chen, Q., Wang, W., Wu, F., De, S., Wang, R., Zhang, B., & Huang, X. (2019). A survey on an emerging area: Deep learning for smart city data. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *3*(5), 392-410.

[25]. Choi, W., Shah, P., & Das, S. K. (2004, August). A framework for energy-saving data gathering using two-phase clustering in wireless sensor networks. In *The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004.* (pp. 203-212). IEEE.

[26]. Chowdhary, R. R., Chattopadhyay, M. K., & Kamal, R. (2020). IoT-Based State of Charge and Temperature Monitoring System for Mobile Robots. In *Innovations in Electronics and Communication Engineering* (pp. 401-413). Springer, Singapore.

[27]. European Commission (EC). (2010). Europe 2020: A strategy for smart, sustainable and inclusive growth. *Working paper {COM (2010) 2020}*.

[28]. Da Xu, L., He, W., & Li, S. (2014). Internet of things in industries: A survey. *IEEE Transactions on industrial informatics*, *10*(4), 2233-2243.

[29]. Keller, T. (2011). *Mining the Internet of Things-Detection of False-Positive RFID Tag Reads Using Low-Level Reader Data* (Doctoral dissertation).

[30]. MacQueen, J. (1967, June). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability* (Vol. 1, No. 14, pp. 281-297).

[31]. Safavian, S. R., & Landgrebe, D. (1991). A survey of decision tree classifier methodology. *IEEE transactions on systems, man, and cybernetics*, *21*(3), 660-674.

[32]. Masciari, E. (2007, September). A Framework for Outlier Mining in RFID data. In *11th International Database Engineering and Applications Symposium (IDEAS 2007)* (pp. 263-267). IEEE.

[33]. Knorr, E. M., & Ng, R. T. (1997, August). A Unified Notion of Outliers: Properties and Computation. In *KDD* (Vol. 97, pp. 219-222).

[34]. Fleury, A., Vacher, M., & Noury, N. (2009). SVM-based multimodal classification of activities of daily living in health smart homes: sensors, algorithms, and first experimental results. *IEEE transactions on information technology in biomedicine*, *14*(2), 274-283.

[35]. Koperski, K., & Han, J. (1995, August). Discovery of spatial association rules in geographic information databases. In *International Symposium on Spatial Databases* (pp. 47-66). Springer, Berlin, Heidelberg.

[36]. Han, J., Cheng, H., Xin, D., & Yan, X. (2007). Frequent pattern mining: current status and future directions. *Data mining and knowledge discovery*, *15*(1), 55-86.

[37]. Zhao, Q., & Bhowmick, S. S. (2003). Sequential pattern mining: A survey. *ITechnical Report CAIS Nayang Technological University Singapore*, *1*(26), 135.

[38]. Agrawal, R., Imieliński, T., & Swami, A. (1993, June). Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on Management of data* (pp. 207-216).

[39]. Bekri, F. E., & Govardhan, A. (2011). Association of Data Mining and healthcare domain: Issues and current state of the art. *Global Journal of Computer Science and Technology*.

[40]. Ranka, S., & Singh, V. (1998, August). CLOUDS: A decision tree classifier for large datasets. In *Proceedings of the 4th Knowledge Discovery and Data Mining Conference* (Vol. 2, No. 8).

[41]. Brunner, S., Kucera, M., & Waas, T. (2017, June). Ontologies used in robotics: A survey with an outlook for automated driving. In *2017 IEEE International Conference on Vehicular Electronics and Safety (ICVES)* (pp. 81-84). IEEE.

[42]. Dhillon, I. S., Guan, Y., & Kulis, B. (2004, August). Kernel k-means: spectral clustering and normalized cuts. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 551-556).

[43]. Ding, K., & Jiang, P. (2017). RFID-based production data analysis in an IoT-enabled smart job-shop. *IEEE/CAA Journal of Automatica Sinica*, *5*(1), 128-138.

[44]. Ester, M., Kriegel, H. P., Sander, J., & Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd* (Vol. 96, No. 34, pp. 226-231).

[45]. Gole, S., & Tidke, B. (2015, January). Frequent Itemset Mining for Big Data in social media using ClustBigFIM algorithm. In *2015 International Conference on Pervasive Computing (ICPC)* (pp. 1-6). IEEE.

[46]. Gu, T., Wang, L., Chen, H., Tao, X., & Lu, J. (2011). Recognizing multiuser activities using wireless body sensor networks. *IEEE transactions on mobile computing*, *10*(11), 1618-1631.

[47]. Guo, H., Liu, J., & Zhao, L. (2017). Big data acquisition under failures in FiWi enhanced smart grid. *IEEE Transactions on Emerging Topics in Computing*.

[48]. Han, J., Cheng, H., Xin, D., & Yan, X. (2007). Frequent pattern mining: current status and future directions. *Data mining and knowledge discovery*, *15*(1), 55-86.

[49]. Huang, K. Y., Chang, C. H., & Lin, K. Z. (2004, September). Prowl: An efficient frequent continuity mining algorithm on event sequences. In *International Conference on Data Warehousing and Knowledge Discovery* (pp. 351-360). Springer, Berlin, Heidelberg.

[50]. Jensen, S. K., Pedersen, T. B., & Thomsen, C. (2017). Time series management systems: A survey. *IEEE Transactions on Knowledge and Data Engineering*, *29*(11), 2581-2600.

[51]. van Kasteren, T., & Krose, B. (2007). Bayesian activity recognition in residence for elders.

[52]. Lyu, L., Nandakumar, K., Rubinstein, B., Jin, J., Bedo, J., & Palaniswami, M. (2018). PPFA: Privacy preserving fog-enabled aggregation in smart grid. *IEEE Transactions on Industrial Informatics*, *14*(8), 3733-3744.

[53]. Roman, R., Zhou, J., & Lopez, J. (2013). On the features and challenges of security and privacy in distributed internet of things. *Computer Networks*, *57*(10), 2266-2279.

[54]. Chen, L., & Ren, G. (2012). The research of data mining technology of privacy preserving in sharing platform of internet of things. In *Internet of Things* (pp. 481-485). Springer, Berlin, Heidelberg.

[55]. Evans, D., & Eyers, D. M. (2012, November). Efficient data tagging for managing privacy in the internet of things. In *2012 IEEE International Conference on Green Computing and Communications* (pp. 244-248). IEEE.

[56]. Appavoo, P., Chan, M. C., Bhojan, A., & Chang, E. C. (2016, January). Efficient and privacy-preserving access to sensor data for Internet of Things (IoT) based services. In *2016 8th International conference on communication systems and networks (COMSNETS)* (pp. 1-8). IEEE.

[57]. Otgonbayar, A., Pervez, Z., & Dahal, K. (2016, October). Toward anonymizing iot data streams via partitioning. In *2016 IEEE 13th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)* (pp. 331-336). IEEE.

[58]. Tso, R., Alelaiwi, A., Rahman, S. M. M., Wu, M. E., & Hossain, M. S. (2017). Privacy-preserving data communication through secure multi-party computation in healthcare sensor cloud. Journal of Signal Processing Systems, 89(1), 51-59.

[59]. Pérez, S., Rotondi, D., Pedone, D., Straniero, L., Núñez, M. J., & Gigante, F. (2017, July). Towards the CP-ABE application for privacy-preserving secure data sharing in IoT contexts. In International conference on innovative mobile and internet services in ubiquitous computing (pp. 917-926). Springer, Cham.

[60]. Ge, M., Hong, J. B., Guttmann, W., & Kim, D. S. (2017). A framework for automating security analysis of the internet of things. Journal of Network and Computer Applications, 83, 12-27.

[61]. Li, Y., & Li, M. (2017). A privacy protection mechanism for numerical control information in Internet of things. International Journal of Distributed Sensor Networks, 13(8), 1550147717726312.

[62]. S. Kumar, O. Kaiwartya, M. Rathee, N. Kumar and J. Lloret, "Toward Energy-Oriented Optimization for Green Communication in Sensor Enabled IoT Environments," in IEEE Systems Journal, April 2020.

[63]. A. Jaiswal, S. Kumar, O. Kaiwartya, N. Kumar, H. Song and J. Lloret, "Secrecy Rate Maximization in Virtual-MIMO Enabled SWIPT for 5G Centric IoT Applications," in IEEE Systems Journal, doi: 10.1109/JSYST.2020.3036417

[64]. Kashyap, P. K., Kumar, S., Dohare, U., Kumar, V., & Kharel, R.: Green Computing in Sensors-Enabled Internet of Things: Neuro Fuzzy Logic-Based Load Balancing. MDPI Electronics, 8(4), pp. 384-405, 2019.

[65]. P. K. Kashyap, S. Kumar and A. Jaiswal, "Deep Learning Based Offloading Scheme for IoT Networks Towards Green Computing," IEEE International Conference on Industrial Internet (ICII), Orlando, USA, 2019, pp. 22-27.

[66]. Kumar, Kirshna; Kumar, Sushil; Kaiwartya, Omprakash; Cao, Yue; Lloret, Jaime; Aslam, Nauman. 2017. "Cross-Layer Energy Optimization for IoT Environments: Technical Advances and Opportunities" Energies 10, no. 12: 2073.

[67]. Indu Dohare& Karan Singh (2019) PSO-DEC: PSO based deterministic energy efficient clustering protocol for IoT, Journal of Discrete Mathematical Sciences and Cryptography, 22:8, 1463-1475, DOI: 10.1080/09720529.2019.1695898

[68]. Mahendra Kumar Jangir& Karan Singh (2019) HARGRURNN: Human activity recognition using inertial body sensor gated recurrent units recurrent neural network, Journal of Discrete Mathematical Sciences and Cryptography, 22:8, 1577-1587, DOI: 10.1080/09720529.2019.1696552

[69]. SudhanshuMaurya&Kuntal Mukherjee (2018) An energy efficient design of cloud of things (CoT), Journal of Information and Optimization Sciences, 39:1, 319-326, DOI: 10.1080/02522667.2017.1374737

[70]. IntyazAlam&Sushil Kumar (2021) Functionality, privacy, security and rewarding based on fog assisted cloud computing techniques in Internet of Vehicles, Journal

of Discrete Mathematical Sciences and Cryptography, DOI: 10.1080/09720529.2020.1794516

[71]. M. M. Rathore, H. Son, A. Ahmad, A. Paul, and G. Jeon, ‘‘Real-time big data stream processing using GPU with spark over Hadoop ecosystem,’’ Int. J. Parallel Program., vol. 46, no. 3, pp. 630–646, 2017.

[72]. Wen-Fei Xi & Dong-Sheng Li (2017) Algorithm of combined reduction based on scattered point cloud, Journal of Information and Optimization Sciences, 38:7, 1221-1228, DOI: 10.1080/02522667.2017.1367503

[73]. X. Ma, X. Fan, J. Liu, H. Jiang, and K. Peng, ‘‘vLocality: Revisiting data locality for MapReduce in virtualized clouds,’’ IEEE Netw., vol. 31, no. 1, pp. 28–35, Jan./Feb. 2017.

[74]. Audu Musa Mabu, Rajesh Prasad &Raghav Yadav (2020) Mining gene expression data using data mining techniques: A critical review, Journal of Information and Optimization Sciences, 41:3, 723-742, DOI: 10.1080/02522667.2018.1555311

[75]. R. R. Expósito, J. GonzÆlez-Domínguez, and J. Touriæo, ‘‘HSRA: Hadoopbased spliced read aligner for RNA sequencing data,’’ PLoS ONE, vol. 13, no. 7, 2018, Art. no. e0201483.

[76]. Swati Rustogi, Manisha Sharma &SudhaMorwal (2017) TID based data and task parallelism for frequent data mining, Journal of Information and Optimization Sciences, 38:6, 961-970, DOI: 10.1080/02522667.2017.1372143

[77]. G. Adomavicius and A. Tuzhilin, ‘‘Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions,’’ IEEE Trans. Knowl. Data Eng., vol. 17, no. 6, pp. 734–749, Jun. 2005.

[78]. Fan Yang (2018) A hybrid recommendation algorithm–based intelligent business recommendation system, Journal of Discrete Mathematical Sciences and Cryptography, 21:6, 1317-1322, DOI: 10.1080/09720529.2018.1526408

[79]. X. W. Meng, S. D. Liu, Y. J. Zhang, and X. Hu, ‘‘Research on social recommender systems,’’ J. Softw., vol. 26, no. 6, pp. 1356–1372, 2015.

[80]. Desmarais M.C., Nkambou R. (eds) User Modelling, UMAP 2012. Lecture Notes in Computer Science, vol 7379. Springer, Berlin, Heidelberg.

[81]. J. S. Breese, D. Heckerman, and C. Kadie, ‘‘Empirical analysis of predictive algorithms for collaborative filtering,’’ in Proc. 14th Conf. Uncertainty Artif. Intell., 1998, pp. 43–52.

[82]. H. Koohi and K. Kiani, ‘‘A new method to find neighbor users that improves the performance of collaborative filtering,’’ Expert Syst. Appl., vol. 83, pp. 30–39, Oct. 2017.

[83]. H. Koohi and K. Kiani, ‘‘User based collaborative filtering using fuzzy C-means,’’ Measurement, vol. 91, pp. 134–139, Sep. 2016.

[84]. K. F. Xylogiannopoulos, ‘‘From data points to data curves: A new approach on big data curves clustering,’’ in Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM), Aug. 2018, pp. 1–4

[85]. L. G. Cuenca, N. Aliane, E. Puertas, and J. Fernandez, ‘‘Data mining approach for traffic hotspots management: Case of Madrid metropolitan area,’’ in Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC), Maui, HI, USA, Nov. 2018, pp. 2484–2489.

[86]. L. Huang, C. Wang, H. Chao, J. Lai and P. S. Yu, "A Score Prediction Approach for Optional Course Recommendation via Cross-User-Domain Collaborative Filtering," in IEEE Access, vol. 7, pp. 19550-19563, 2019, doi: 10.1109/ACCESS.2019.2897979.

[87]. M. Muzammal, M. Gohar, A. U. Rahman, Q. Qu, A. Ahmad, and G. Jeon, ‘‘Trajectory mining using uncertain sensor data,’’ IEEE Access, vol. 6, pp. 4895–4903, 2018.

[88]. H. Liu, ‘‘Resource recommendation via user tagging behavior analysis,’’ Cluster Comput., vol. 22, pp. 885–894, Jan. 2019.

[89]. B. K. Patra, R. Launonen, V. Ollikainen, and S. Nandi, ‘‘A new similarity measure using Bhattacharyya coefficient for collaborative filtering in sparse data,’’ Knowl.-Based Syst., vol. 82, pp. 163–177, Jul. 2015.

[90]. C. Wang, Z. Deng, J. Lai and P. S. Yu, "Serendipitous Recommendation in E-Commerce Using Innovator-Based Collaborative Filtering," in IEEE Transactions

on Cybernetics, vol. 49, no. 7, pp. 2678-2692, July 2019, doi: 10.1109/TCYB.2018.2841924.

[91]. C. Tran, J. Kim, W. Shin and S. Kim, "Clustering-Based Collaborative Filtering Using an Incentivized/Penalized User Model," in IEEE Access, vol. 7, pp. 62115-62125, 2019, doi: 10.1109/ACCESS.2019.2914556.

[92]. S. Ajoudanian and M. N. Abadeh, "Recommending human resources to project leaders using a collaborative filtering-based recommender system: Case study of gitHub," in IET Software, vol. 13, no. 5, pp. 379-385, 10 2019, doi: 10.1049/iet-sen.2018.5261.

[93]. Akyildiz IF, Su W, Sankarasubramaniam Y and Cayirci E, Wireless sensor networks: a survey, Computer Networks, Vol.38, No. 4, pp. 393-422, 2002.

[94]. Kashyap, P. K., Kumar, S., and Jaiswal, A.: Deep Learning Based Offloading Scheme for IoT Networks Towards Green Computing. IEEE International Conference on Industrial Internet (ICII), pp. 22-27, Orlando, FL, USA, 2019.

[95]. F. Bouabdallah, N. Bouabdallah, and R. Boutaba, “Towards reliable and efficient reporting in wireless sensor networks,” IEEE Trans. MobileComput., vol. 7, no. 8, pp. 978–994, Aug. 2008.

[96]. Ishmanov F, Malik AS and Kim SW, Energy consumption balancing (ECB) issues and mechanisms in wireless sensor networks (WSNs): A comprehensive overview, European Transactions on Telecommunications, Vol. 22, pp. 151-167, 2011.

[97]. Ahmed AA and Mohammed Y, A survey on clustering algorithms for wireless sensor networks, Elsevier, ComputerCommunications, Vol. 30, pp. 2826-2841, 2007.

[98]. J. Zhou, H. Jiang, J. Wu, L. Wu, C. Zhu, and W. Li, ‘‘SDN-based application framework for wireless sensor and actor networks,’’ IEEE Access,vol. 4, pp. 1583–1594, 2016.

[99]. C. Guestrin, P. Bodik, R. Thibaux, M. Paskin, and S. Madden, ‘‘Distributedregression: An efficient framework for modeling sensor network data,’’ inProc. 3rd Int. Symp. Inf. Process. Sensor Netw., Apr. 2004, pp. 1–10.

[100]. Kashyap, P.K, Kumar, S. “Genetic-fuzzy based load balanced protocol for WSNs” International Journal of Electrical and Computer Engineering, Vol. 9, No.2, April 2019, pp.1168-1183.

[101]. R. S. Sutton and A. G. Barto, Reinforcement Learning: An Introduction, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.

[102]. T. Hu and Y. Fei, ‘‘QELAR: A machine-learning-based adaptive routing protocol for energy-efficient and lifetime-extended underwater sensornetworks,’’ IEEE Trans. Mobile Comput., vol. 9, no. 6, pp. 796–809,Jun.2010.

[103]. N. Javaid, O. A. Karim, A. Sher, M. Imran, A. U. H. Yasar, and M. Guizani,‘‘Q-learning for energy balancing and avoiding the void hole routing protocol in underwater sensor networks,’’ in Proc. 14th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC), Jun. 2018, pp. 702–706 Singh, M & Prasanna, V (2003).

[104]. W. Guo, C. Yan, and T. Lu, “Optimizing the lifetime of wireless sensor networks via reinforcement-learning-based routing,” Int. J. Distrib. Sensor Netw., vol. 15, no. 2, 2019,

[105]. Z. Jin, Y. Ma, Y. Su, S. Li, and X. Fu, “A Q-learning-based delay-aware routing algorithm to extend the lifetime of underwater sensor networks,” Sensors, vol. 17, no. 7, 2017.

[106]. J.R. Douceur , A. Adya , W.J. Bolosky , D. Simon , M. Theimer , “Reclaiming space from duplicate files in a serverless distributed file system,” Proc. of International Conference on Distributed Computing Systems, 2002, pp. 617–624 .

[107]. O. Regev, “On Lattices, Learning with Errors, Random Linear Codes, and Cryptography,” JACM, vol. 56, no. 6, article 34, 2009.

[108]. Hur, Junbeom, et al. “Secure data deduplication with dynamic ownership management in cloud storage,” IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 11, 2016, pp. 3113-3125.

[109]. J. Li, J. Li, D. Xie, and Z. Cai, ‘‘Secure auditing and deduplicating data in cloud,’’ IEEE Trans. Comput., vol. 65, no. 8, pp. 2386–2396, Aug. 2016.

[110]. Yan, Jiajun, et al. “Secure and efficient big data deduplication in fog computing,” Soft Computing, 2019, pp. 1-12.

[111]. Yan, Yunxue, et al. “A dynamic integrity verification scheme of cloud storage data based on lattice and Bloom filter,” Journal of information security and applications, vol. 39, 2018, pp. 10-18.

[112]. Yuan, Haoran, et al. “DedupDUM: Secure and scalable data deduplication with dynamic user management,” Information Sciences, vol. 456, 2018, pp. 159-173.

[113]. D. X. Song, D. Wagner, and A. Perrig, “Practical techniques for searches on encrypted data,” Proc. IEEE Symposium on Security and Privacy (S&P’00), 2000, pp. 44–55.

[114]. Dong, Changyu, Giovanni Russello, and Naranker Dulay, “Shared and searchable encrypted data for untrusted servers.” Journal of Computer Security, vol. 19, no. 3, 2011, pp. 367-397.

[115]. Sun, Wenhai, et al, “Protecting your right: Verifiable attribute-based keyword search with fine-grained owner-enforced search authorization in the cloud,” IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 4, 2014, pp. 1187-1198.

[116]. Huang, Cheng, et al. “Reliable and Privacy-Preserving Selective Data Aggregation for Fog-Based IoT,” 2018 IEEE International Conference on Communications (ICC), 2018, pp.1-6.

[117]. www.microsoft.com/en-us/download/details.aspx?id=52371.

[118]. Srinivas, J., et al. "Cloud centric authentication for wearable healthcare monitoring system." IEEE Transactions on Dependable and Secure Computing (2018).

[119]. Srinivas, J., et al. "Cloud centric authentication for wearable healthcare monitoring system." IEEE Transactions on Dependable and Secure Computing (2018).