Montclair State University
Montclair State University Digital Commons

5-2019

A Privacy-Preserving Framework for Collaborative Association Rule Mining in Cloud

Salha Albehairi
Montclair State University

Follow this and additional works at: https://digitalcommons.montclair.edu/etd

Part of the Information Security Commons

Abstract

Collaborative Data Mining facilitates multiple organizations to integrate their datasets and extract useful knowledge from their joint datasets for mutual benefits. The knowledge extracted in this manner is found to be superior to the knowledge extracted locally from a single organization's dataset. With the rapid development of outsourcing, there is a growing interest for organizations to outsource their data mining tasks to a cloud environment to effectively address their economic and performance demands. However, due to privacy concerns and stringent compliance regulations, organizations do not want to share their private datasets neither with the cloud nor with other participating organizations. In this paper, we address the problem of outsourcing association rule mining task to a federated cloud environment in a privacy-preserving manner. Specifically, we propose a privacy-preserving framework that allows a set of users, each with a private dataset, to outsource their encrypted databases and the cloud returns the association rules extracted from the aggregated encrypted databases to the participating users. Our proposed solution ensures the confidentiality of the outsourced data and also minimizes the users' participation during the association rule mining process. Additionally, we show that the proposed solution is secure under the standard semi-honest model and demonstrate its practicality.

MONTCLAIR STATE UNIVERSITY

**A Privacy-Preserving Framework for Collaborative
Association Rule Mining in Cloud**

by

Salha Albehairi

A Master's Thesis Submitted to the Faculty of

Montclair State University

In Partial Fulfillment of the Requirements

For the Degree of Master of Science

May 2019

College: College of Science and
Mathematics

Department: Computer Science

Thesis Committee:

Dr. Bharath K. Samanthula
Thesis Sponsor
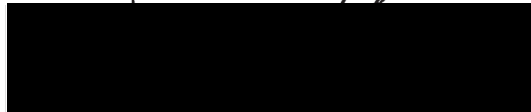
Dr. Boxiang Dong
Committee Member

Dr. Kazi Zakia Sultana
Committee Member

2

A PRIVACY-PRESERVING FRAMEWORK FOR COLLABORATIVE ASSOCIATION RULE MINING IN CLOUD

A THESIS

Submitted in partial fulfillment of the requirements for the degree of Master of Science

by

SALHA ALBEHAIRI

Montclair State University

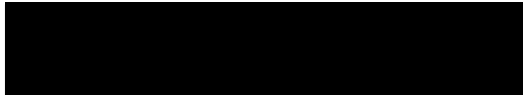May 2019

Acknowledgements

First of all, thanks to ALLAH for giving me full strength and ability to complete this thesis. I also would like to express my sincere gratitude to my advisor Prof. Bharath K. Samanthula for the continuous support of my thesis study and research, for his patience, enthusiasm, guidance, assistance, and immense knowledge. He spends vary much time instructing me how to write a paper, how to search literature and how to organize data. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my thesis.

Besides my advisor, I would like to thank my thesis committee: Dr. Boxiang Dong, and Dr. Kazi Zakia Sultana for spending time to read this thesis and providing useful suggestions about this thesis.

Nobody has been more important to me in the pursuit of this thesis than the members of my family. I wish to thank my loving and supportive husband, Abdullah, and my wonderful son Yazn, who provide unending inspiration. Most importantly, I would like to thank my parents; whose love and guidance are with me in whatever I pursue. They are the ultimate role models.

During the period of completing my master's degree, my sisters, brothers, friends are helpful to color my life. I have to acknowledge all of them: Fatimah, Ghala, Aiosh, Maryam, Bashayer, Boshra, Saad, Jojo, Zezo, Somah, Sema, and Amani.

Table of Contents

List of Figures

List of Tables

1   Introduction

1.1   Background

With the rapid growth of processing and storage technologies, internet enabled organizations and companies store massive amounts of data. Thus, the amount of data in the world doubles every 20 months and the size and number of databases are increasing too [1]. However, extracting useful information becomes more challenging because of the huge amount of data. This technological trend has enabled the realization of a new computing model of cloud computing, where all resources are provided as utilities to end users [2]. NIST's definition of cloud computing: cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [3]. There are many technologies that are working together to give us an overview of the cloud.

- **Service Models** There are different types of service models. The following are the most popular models [4]:

    - **Infrastructure-as-a-Service (IaaS)**

      IaaS is a form of cloud computing that provides virtualization resources such as storage, network, and server. IaaS provider also provides other services to accompany those infrastructure components. These can include detailed billing, monitoring, log access, security, as well as storage resiliency, such as backup, replication, and recovery. IaaS is more cost-efficient because the business does not have to buy, manage, and support the underlying infrastructure. Examples are EC2, GCE, and Azure.

8

– **Platform-as-a-Service (PaaS)**

PaaS is a cloud computing model in which the cloud service provider offers hardware and software tools for the end user to employ them for application development over the Internet. Some examples of PaaS are Google App Engine and Microsoft Windows Azure.

– **Software-as-a-Service (SaaS)**

SaaS is a cloud computing model in which the cloud service provider offers host applications over the Internet and makes it available to the consumer. The main benefit of SaaS is to remove the need for the organizations or companies to install the applications on their own computers.

– **Data Mining as a Service (DMaaS)**

DMaaS is a software and computing infrastructure that allows interactive mining of scientific data in the cloud. It enables users to used cloud for advancing data analyses.

- **Enabling Technologies**

Utility computing is one of the fundamental concepts of cloud computing. CPU, memory, and storage are provided to the end user as a utility service. The end-user pays for the utilities that are consumed [4].

- **Important Features of Cloud Service Providers**

There are many components that are provided from Cloud Service Providers (CSPs) like Infrastructure as a Service (IaaS), Software as a Service (SaaS) and Platform as a Service (PaaS) to business organizations or individuals. In addition, there are many cloud service providers available these days. The consumer selects CSP to depend

on many criteria such as its requirements, budget, and security. This section presents important factors before consumer selects CSP [4].

– **Reliability**

Businesses are concerned about reliability because when service goes down, an outage can impact the business significantly. So, the cloud storage service level agreement specifies the level of reliability like 99.99999% availability. Also, they should be concerned about the time for recovery and how CSP will protect the data

– **Customer Support Services**

Support services are important factors since the consumer needs support services available at all times. When the organization requires support services in any situation, 24/7 real-time support for solving any problem is essential.

– **Security**

The disadvantage of moving to a cloud service is that it is less secure. Security can be compromised when cloud service providers do not have successful control over malware and threat protection, encryption techniques, government rules and regulations etc.

– **Manageability**

Cloud service providers should make managing the server for deployment easy to attract its users. So, the ability to manage the system for a long time is an important factor in selecting a CSP.

• **Data Mining in Cloud**

Data mining in cloud is a process of extracting structure information from unstruc-

tured or semi-structured data sources and finding helpful patterns or relationships in a group with a massive amount of data and gain knowledge of the pattern [5]. There are some techniques that are used in cloud for data mining, here we discus some of them:

- Clustering is a process of making similar abstract objects into classes of similar features, so each class can be treated as one group. This type of technique can be used on market research, pattern recognition, and data analysis [6].

- Classification is a process of discovering a model that characterizes and identifies data classes and concepts such as finding low/ high value [7].

- Association Rules is a process of focusing on finding interesting relations between items in massive amount of data. It plays the important part of shopping basket to predict customer behavior. For example, if a customer buys eggs, he is 80% likely to purchases bread [8].

- Regression is a technique for predicting range of numeric values or outcome. For example, using regression to predict the cost of a product [5].

1.2 Motivation

Most organizations today use cloud environments for storage and data processing. For using the full features of cloud computing, external cloud providers transfer, process, store, and retrieve the data. Data owners are very concerned about confidentiality, integrity, security, and techniques of mining the data from the cloud, so they are very suspicious of placing their data outside their own control [9]. Thus, one of the main issues is that the server of the cloud provider has access to the data and that may affect the privacy of sensitive information such as looking at the transaction database; this can indicate the products

purchased, and the most frequent item-set, and then know the mined encrypted patterns, which is called Association rule mining [9].

In marketing, users can benefit from data mining as they increase their profits by sharing and analyzing results with other users to gain a large sample size of transactions. Thus, the results will be more accurate. This type of information requires more privacy. Our approach confirms that the association rules remain secure when the owners of data used a federated cloud to outsource their encrypted data. Homomorphic encryption is applied on our framework, which is based on computing over encrypted data to ensure confidentiality. Thus, the data owners first outsource their encrypted data to a federated cloud. Then, the federated cloud securely computes the frequent item-set and association rules mining within collaborative cloud environment while minimizing users' intervention. Then, the federated cloud returns the correct results to end users. So, our proposed model is more secure than existing models.

1.3 Problem Setting

Our proposed model is based on two types of entities: U_1......, U_n which indicates n data owners because our framework is a multi-tenant environment, and C_1, C_2 which indicate cloud service providers which are implementing collaborative computations on encrypted data. Each user U_i has a transactional database, indicated by T_1, T_2..., T_n. Encrypted transactions indicated by $E_{pk}(T_1)$, $E_{pk}(T_2)$..., $E_{pk}(T_n)$ which is sent by each user U_i to cloud. Figure 1 presents our model. In our approach, unauthorized users and cloud servers cannot have access to original data. Also, we suppose that the two cloud service providers are semi-honest (or honest-but-curious), which means all of them accurately follow the protocol using its correct input. The semi-honest model is more efficient than those under
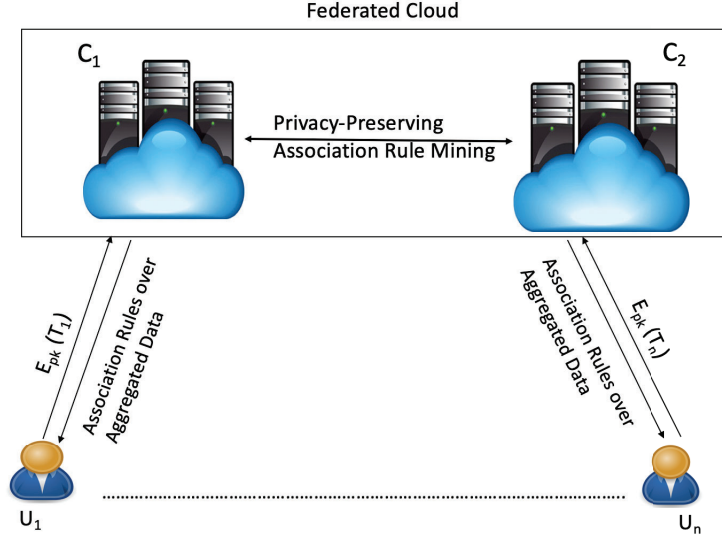
Figure 1: Proposed Model

the malicious adversary model [10]. First, each data owner has to encrypt their database and then outsource it to a federated cloud environment. Then, C_1 and C_2 securely compute the frequent item-sets and association rule mining based on a homomorphic encryption technique, which means securely computing the result and sending it to end user U_i. Thus, our proposed model focuses on: (a) the federated cloud securely computing the frequent item-sets over encrypted database of different users, and (b) the federated cloud securely computing the association rules mining and returning the correct result to end users.

1.4 Main Contributions

The most important contributions of our model are:

- **Data Confidentiality**: cloud servers and an unauthorized user cannot have access to user's database.

- **Correct Result**: our algorithm returns correct results of the association rule mining

to the end users and this result is same as the result of the standard algorithm.

- **Reduced User's Participation**: users do not need to stay online until the mining process is finished. When the users send their encrypted database to the cloud, it securely computes the frequent item-sets and association rules mining and returns the result to them. Thus, it minimizes user intervention.

1.5   Organization

The rest of the thesis is organized as follows. Section 2 presents an overview of related work in the same area. Section 3 presents preliminary techniques of our protocol. Section 4 explains our proposed approach. Section 5 describes the implementation and experimental results. Section 6 concludes and discusses future work.

2 Related Work

2.1 Association Rule Mining

There are some approaches proposed for performing association rule mining on the data in the cloud. Wong et al. [11] proposed one to n items mapping by adding fake items into transactions database. That means, the data owner adds fake items F to the dictionary and then maps each item x to random subset of F. Thus, the server cannot easily find out the distribution of the correct item-sets in the database. This approach has two weaknesses: first, the probability of adding fake items to each transaction is the same. Thus, the fake transactions will appear in a large database with similar frequencies; second, the fake items added to transactions independently of the items are already present, and thus with some calculations the fake items can be removed and the true items can be identified. Another approach is proposed by Tai et al. [12]. The basic idea of this approach is based on K-support anonymity to protect each sensitive item with $k-1$ on the database and other items with the same support value. Thus, each transformed item cannot be identified from at least $k-1$ other items. This approach used pseudo taxonomy tree to limit the occurrence of fake items. Their used K-support anonymity to protect the sensitive information.

2.2 Privacy Preserving Data Mining in Cloud

There are some approaches proposed for privacy preserving data mining in cloud. Based on the K-support anonymity idea, Ginannotti et al. [13] proposed a model that extended the concept of K-support anonymity to K-privacy. So, each transformed item-set cannot be identified from at least $k-1$ other item-sets with the same size. To achieve K-privacy, there were three steps. First, they used one to one mapping approach for replacing plain items.

Second, they put items on groups for K-privacy. Third, they added fake transactions. Recently, Xun Yi et al. [14] proposed techniques for protecting the outsource data and association rules mined. These techniques are K-anonymity, K-support, and K-privacy. After data owner outsource, his/her encrypted data to cloud, data owner sends the task to $n \geq 2$ semi-honest servers to do association rule mining on encrypted data and return the encrypted result to the user. This solution is based on distributed ELGamal cryptosystem to reach the privacy of item, transaction, and database. The weakness of this solution is the possibility of compromising all servers [14]. Also, Xun Yi et al. proposed three techniques: association rule mining on encrypted item-set with item privacy, association rule mining with transaction privacy, association rule mining with data privacy. Here, we discuss them in details.

1. Association Rule Mining on Encrypted Item-set with Item Privacy

   To compute the association rule mining, the user first chooses the minimum number of threshold and support to send them to the database server. Then, applying an Apriori algorithm on the encrypted transaction database, based on threshold and support values; encrypted association rules mining is returned to the user. The limitation of this method is that the database server knows the minimum support and threshold, so it does not remain secure [14].

2. Association Rule Mining with Transaction Privacy

   In this solution, the client sends to a DM server the minimum encrypted value of support. This solution ignored confidence. It assumed that the DM server and n DM servers collaborate together to mine all the encrypted frequent item-sets and return to the client encrypted supports. After decryption, the client will have strong association rules that satisfy the minimum supports and confidence [14]. To build noisy

transactions, the DM servers choose and encrypt some items and then put these encrypted items into the noisy transaction. This way is difficult to remove the effect of noisy transactions without decryption with minimum frequent item-sets. To get a list of encrypted items, the DB server and the n DM servers participate to anonymize the dictionary of all possible items. By using the same item identification algorithms, the encrypted items in the transaction database are replaced with encrypted items in the dictionary, but the content of the original transaction is not changed. Then, it applies the Apriori algorithm on encrypted the item-set. Thus, the original transactions and noisy transactions are mixed together. This technique has item and transaction privacy but it leaks some information about the original transaction because of the item identification algorithm, so it is not secure [14].

3. Association Rule Mining with Data Privacy

This method makes the database server divide the database into n subsets. After computing the frequent item-set, it returns the result to the user if the result is 1, which means there is a frequent item-set. If it is zero, there is no frequent item-set. This solution leaks some information about frequent item-sets [14].

In our paper, we focus on the problem of outsourcing the association rule mining task to a federated cloud environment because most previous solutions make the result of mined rule shared with other parties. Thus, the difference between our solution and previous solutions is that the basic information and mined result stay secure and private to the end user and only has a small amount of data leakage.

3 Preliminaries

3.1 Apriori algorithm

The Apriori algorithm developed by [15] is a big achievement in the history of association rules mining. It is the most well-known association rule algorithm. The Apriori algorithm generates the candidate item-sets to be counted in a pass by using only the large item-sets found in the previous pass without looking at the transactions in the database. This algorithm uses the property of a large item-set having to be a large item-set for any subset [15].

3.2 Homomorphic Encryption

Homomorphic Encryption is a method of encryption that uses computation on encrypted data without decrypting them first. We will focus on two efficient schemes. First, the El-Gamal scheme is a public-key encryption algorithm based on the Diffie–Hellman key exchange [16]. Second, the Pailler scheme has homomorphic Addition propriety that means the product of two cipher-texts will decrypt to the sum of their corresponding plain-texts. Also, it has Homomorphic Multiplication of plain-texts that means an encrypted plaintext raised to the power of another plaintext will decrypt to the product of the two plain-texts. However, given the Paillier encryptions of two values, there is no way to compute an encryption of the product of these values without knowing the private key [16].

3.2.1 Paillier Cryptosystem

The Paillier cryptosystem is an additive homomorphic and a probabilistic asymmetric algorithm for public key cryptography [17]. Suppose E_{pk} is the encryption function with public

key pk given by (N, g), where N is a product of two large primes, and g is in $Z_{N^2}^*$. Also, the decryption function with secret key sk is D_{sk}. For given $a, b \in Z_N$, the Paillier encryption scheme offer the following properties:

- **Homomorphic Addition**

  $E_{pk}(a + b) \leftarrow E_{pk}(a) \bullet E_{pk}(b) \mod N^2$

- **Homomorphic Multiplication**

  $E_{pk}(ab) \leftarrow E_{pk}(a)^b \mod N^2$

3.3 Cryptographic Primitives

- **Secure Multiplication**

  In this protocol, $(E_{pk}(a), E_{pk}(b))$ is considered as private input of a party P_1 and outputs $E_{pk}(a \bullet b)$ to P_1, where a and b are not known to P_1 and P_2. In this process, no information regarding a and b is leaked to P_1 or P_2. The final output $E_{pk}(a \bullet b)$ is known only to P_1 [18]. The basic idea of SM protocol is based on the following property for any given a, b $\in Z_N$: $a \bullet b = (a + r_a) \bullet (b + r_b) – a \bullet r_b – b \bullet r_a – r_a \bullet r_b$

- **Secure Comparison**

  In this protocol, P_1 has encryption value of a and b and P_2 has the secret key. Now, P_1 and P_2 have to apply a secure comparison. Suppose we have $((a) \geq (b))$ as a condition. After applying SC, if the output is one, that means the condition is true. If the output is zero, that means the condition is false. The final output is known only to P_1 [19].

4 Proposed Algorithm

4.1 Data Outsourcing

In the beginning, the users securely outsourced their transaction database using randomization approach where each user U_i randomized each item x and send the values to C_1 and C_2 such that $x = x_1 + x_2 \mod N$. Thus, the user U_i sends the random value x_1 to C_1 and x_2 to C_2. When C_2 receives its random value x_2, it will encrypt and send it to C_1. Then, C_1 using secure addition, which uses the additive homomorphic property of Paillier Cryptosystem to combine the received value with its random value x_1 to find the encrypted value of x by this equation: $E_{pk}(x_1) \bullet E_{pk}(x_2) = E_{pk}(x_1 + x_2 \mod N) = E_{pk}(x)$. At this time, only C_1 knows the encrypted transaction database and threshold (α) that is sent by the user to it. The value of (α) represents the minimum value of support. The support value determines the number of times the rule is applied to a given data set. The importance of calculation support is to avoid rules that have low value of support because these rules might happen by chance. Our proposed model contains two main phases: (a) secure computation of frequent item-sets, and (b) secure computation of association rules mining and returning results to end user. We discuss these two phases in the details on following section.

4.2 Secure Computation of Frequent Item-sets

Now, after we have encrypted data and the value of threshold (α) from user, C_1 can securely compute the frequent item-set based on the addition property of Paillier cryptosystem. For example, suppose we have item i_1 and i_2 in the transactions, and they have these values (1,0,1), (0,1,1). Then, C_1 uses secure addition for all values of i_1, $E_{pk}(1) \bullet E_{pk}(0) \bullet E_{pk}(1)$ equal to $E_{pk}(1 + 0 + 1)$, the result will be 2 for both i_1 and i_2. After that, C_1 and C_2

collaborate to do secure comparison with the threshold (α). If the result equal or greater than threshold (α), 1 will be sent to C_1, else 0 will be sent. In our example, assume that (1,1) are sent to C_1. Then, C_1 and C_2 collaborate to do secure multiplication of the values of the pair (i_1, i_2): (1 $\bullet$ 0), (0 $\bullet$ 1), (1 $\bullet$ 1) and then C_1 computes the result using secure addition again. Finally, using the secure comparison protocol [19], if the result is greater than or equal to (α), (i_1, i_2) will be added to the frequent item-sets. All of the steps in our proposed solution are mentioned in algorithm 1. Here, we will discuss an example of how to compute frequent item-set. Suppose we have sample transaction database D:

Table 1: Sample Transaction Database

	i_1	i_2	i_3	i_4
T_1	0	0	0	1
T_2	1	0	1	0
T_3	1	0	1	0
T_4	1	1	1	1

Assume that the user U_i sets the value of threshold (α) to 2. First, computing the frequent K-itemsets by using Secure Additive property of Paillier cryptosystem:

- $i_1 = E_{pk}(0) \bullet E_{pk}(1) \bullet E_{pk}(1) \bullet E_{pk}(1)$

  $= E_{pk}(0 + 1 + 1 + 1)$

  $= E_{pk}(3)$

- $i_2 = E_{pk}(0) \bullet E_{pk}(0) \bullet E_{pk}(0) \bullet E_{pk}(1)$

  $= E_{pk}(0 + 0 + 0 + 1)$

  $= E_{pk}(1)$

- $i_3 = E_{pk}(0) \bullet E_{pk}(1) \bullet E_{pk}(1) \bullet E_{pk}(1)$

  $= E_{pk}(0 + 1 + 1 + 1)$

  $= E_{pk}(3)$

- $i_4 = E_{pk}(1) \bullet E_{pk}(0) \bullet E_{pk}(0) \bullet E_{pk}(1)$

  $= E_{pk}(1 + 0 + 0 + 1)$

  $= E_{pk}(2)$

Table 2: K_{th} Iteration of Database

	i_1	i_2	i_3	i_4
T_1	0	0	0	1
T_2	1	0	1	0
T_3	1	0	1	0
T_4	1	1	1	1

During the next step, C_1 and C_2 collaborate to compute the frequent item-set based on secure comparison protocol. This protocol compares the results from i_1 to i_4 with $E_{pk}(\alpha)$. For example, i_1 equal to 3, this encrypted value compares to (α) if the result is one (true), that means 3 greater than or equal the value of (α). Then, the result 0 (false) or 1(true) will be sent to C_1. In our example, 1,0,1,1 will be sent to C_1. That means i_2 is not a frequent item-set because $E_{pk}(1)$ is less than the value of threshold (α). Then, federated cloud find the frequent item-set for (k+1) from the first iteration: (i_1, i_2), (i_2, i_3), (i_3, i_4), (i_1, i_3), (i_2, i_4), (i_1, i_4). Since i_2 is not the frequent item-set, it will be removed from the second iteration. Thus, the second iteration will be as follows: (i_1, i_3), (i_3, i_4), (i_1, i_4).

Table 3: K+1 Iteration of (i_1, i_3)

	i_1	i_3
T_1	0	0
T_2	1	1
T_3	1	1
T_4	1	1

Then, computing the secure multiplication is as follow:

- $T_1 = E_{pk}(0 \bullet 0) = E_{pk}(0)$

- $T_2 = E_{pk}(1 \bullet 1) = E_{pk}(1)$

- $T_3 = E_{pk}(1 \bullet 1) = E_{pk}(1)$

- $T_4 = E_{pk}(1 \bullet 1) = E_{pk}(1)$

  Then, applying the additive property of Paillier cryptosystem again will give us E_{pk} (3) which is greater than $E_{pk}(\alpha)$ by using the comparison protocol. Thus, (i_1, i_3) is the frequent item-set. Then, applying the same steps on (i_3, i_4) to determine if it is a frequent item-set or not.

Table 4: K+1 Iteration of (i_3, i_4)

	i_3	i_4
T_1	0	1
T_2	1	0
T_3	1	0
T_4	1	1

- $T_1 = E_{pk}\,(0 \bullet 1) = E_{pk}\,(0)$

- $T_2 = E_{pk}(1 \bullet 0) = E_{pk}\,(0)$

- $T_3 = E_{pk}\,(1 \bullet 0) = E_{pk}(0)$

- $T_4 = E_{pk}\,(1 \bullet 1) = E_{pk}(1)$

Then, applying additive property $E_{pk}(0 + 0 + 0 + 1)$ will give us $E_{pk}(1)$. Based on the comparison protocol, this value is less than $E_{pk}(\alpha)$. That means (i_3, i_4) is not the frequent item-set. So, we have (i_1, i_3) as frequent item-set.

4.3 Secure Computation of Association Rules

After finding the frequent item-sets, the association rules are computed to discover interest relationships in large item-sets. The association rules are an expression term of the form $A \rightarrow B$, where A and B are separated item-sets, i.e., $A \cap B = \emptyset$. The power of an association rule can be determined by support and confidence. We calculated support on the first stage. Here, we calculate the minimum value of confidence (β) which is determine how often items in B appear in transactions containing A. We can calculate the value by this equation: Confidence $(A \rightarrow B) =$

$$\frac{Frequency(A, B)}{Frequency(A)}$$

To compute all possible rules, C_1 first generates empty set, and then computes all values independently. Next, it computes the confidence of ($A \rightarrow B$) by converting the equation to

24

Algorithm 1

Require:

C_1 has encrypted database, and C_2 has encrypted keys. Threshold (α) the minimum value of confidence, threshold (β) the minimum value of support, and x_i indicates frequent i-item-sets

1: **for** $(x_i$ to $i)$ **do**
2:     **if** (i=1) **then**
3:         **for** $y \in$x$_i$ **do**
4:            Secure Addition(y)
5:            Secure Comparison(y) with (α)
6:            **if** (Result= 1) **then**
7:                Send 1 to C_1
8:                Add the result to frequent item-sets F
9:            **else**
10:                Send 0 to C_1
11:     **else**
12:         **for** $y \in$x$_i$ **do**
13:            Secure Multiplication(y)
14:            Secure Addition(y)
15:            Secure Comparison(y) with (α)
16:            **if** (Result =1) **then**
17:                Send 1 to C_1
18:                Add the result to frequent item-sets F
19: C_1 initializes the item-sets of association rules R to $\emptyset$
20: **for** $X \in F$ **do**
21:     Secure Multiplication(x)
22:     Secure Comparison(x) with (β)
23:     **if** (Result= 1) **then**
24:         rule is added to the empty set R
25: C_1 sends encrypted R to the user

Figure 2: Algorithm for Proposed Model

linear form $(Freq(A, B)) \geq (Freq(A) * (\beta))$. Then, C_1 and C_2 collaborate to apply Secure Comparison and Secure Multiplication algorithms, so if frequency of (A, B) is greater than or equal to frequency of (A) multiply by the value of (β), the result will be 1, that means $(Freq(A, B))$ is greater than $(Freq(A) * \beta)$. Then, $(A \rightarrow B)$ rule is added to the empty set. At the end of this stage C_1 will have the possible rules. After, computing all possible rules, the C_1 sends the encrypted result of association rules to the end user.

Continuing with the previous example to compute association rules. At the end of the first phase we have the frequent item-set (i_1, i_3). Now, we can securely compute the association rules. First, C_1 generates empty-set R. Next, federated cloud compute the confidence of $(i_1 \rightarrow i_3)$ by this equation $(Freq(i_1, i_3)) \geq (Freq(i_1) * (\beta))$. Here on our example, the value of $(Freq(i_1, i_3))$ is 3, and $(Freq(i_1))$ equal to 3, and suppose the value of (β) is 1. Thus, we have $3 \geq (3 \bullet 1)$. After that, C_1 and C_2 apply secure comparison and secure multiplication. The final result will be 1. Then, rule $(i_1 \rightarrow i_3)$ is added to R.

4.4 Security Comparison

The table below shows a comparative analysis on the data protected using each approach.

Table 5: Security Comparison

Solutions	Frequency of Item-set	Minimum Threshold	Database
Item Privacy	χ	χ	χ
Transaction Privacy	χ	✓	χ
Database Privacy	χ	✓	✓
Proposed Solution	✓	✓	✓

We can see from the table above that the first solution proposed by Xun Yi et al. [14] leaks some information about the frequency of item-set, minimum threshold and transac-

tion database. The transaction privacy solution proposed by Xun Yi et al. [14] also leaks some information about the frequency of item-set and transaction database but the threshold remains secure because it is encrypted. The third solution is the database privacy which is also leaks information about the frequency of item-set but protects both minimum threshold and database. However, our proposed model hides support, threshold and database while ensuring privacy which is considered the best approach in the comparative security analysis.

5 Experimental Evaluation

5.1 Experimental Setup

In this section, we will present our experimental results in detail under different parameters. We used the Paillier cryptosystem and implemented our protocol using Java on MacOS machine with 3.3 GHz Intel Core i7 processor with 8 GB of memory. Since it is hard to control real dataset parameters, we generated random datasets rely on different parameters. By using these random datasets, we implemented our protocol with more details of computation costs under different parameters settings. We encrypted these datasets using the Paillier cryptosystem with fixed encryption key size 1024 and stored these encrypted datasets on our machine. Then, we executed different queries on these encrypted datasets.

5.2 User Performance

In figure 3, by varying the number of transactions (n) and the number of attributes (m), we evaluated the computation cost of the encrypted time to outsource the data to cloud for the given user. We can see that the time of optimizing encryption grows linearly with n and m. For example, when $m = 20$, the encryption time increases from 0.26 to 0.34 second when n is varied from 5000 to 10,000.

5.3 Federated Cloud Performance

In figure 4, by fixing $k = 5$ and varying the number of transactions (n) and the number of attributes (m) to find the frequent item-sets, the time also increases linearly with n and m. Figure 5 illustrates the running time for finding the association rules mining from the frequent item-sets. We can see that the running time is increased when the number of

association rules increases. For example, when the number of rules is equal to 1000, the running time is 133 seconds, and when the number of rules is equal to 2000, the running time is 266 seconds.
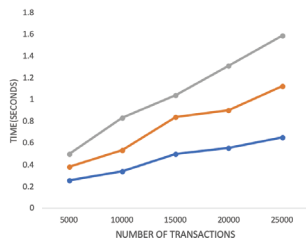

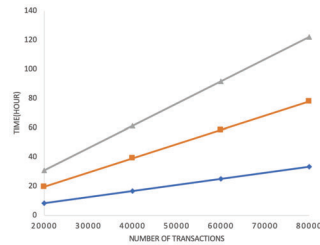Figure 3: User Computation Time
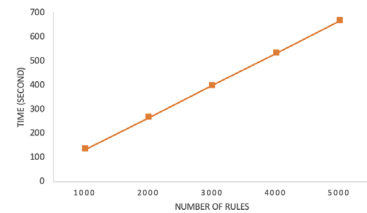

Figure 4: Computation Time of Phase one

Figure 5: Computation Time of Phase Two

6 Conclusions

6.1 Technical Contributions

Our approach confirms that the association rules remain secure when the owners of data used a federated cloud to outsource their encrypted data. It makes the value of the threshold and the encrypted data of transaction remain secure. In addition, the cloud servers or unauthorized users will not be able to know any information about them. To ensure confidentiality, homomorphic encryption is applied on our framework, which is based on computing over encrypted data. Also, we applied the Paillier cryptosystem using Secure Addition, Secure Multiplication, and Secure Comparison to ensure the privacy. Our model is based on two phases: (a) secure computation of frequent item-sets, and (b) secure computation of association rules mining task in collaborative cloud environment. Then, the federated cloud returns the association rules mining on aggregated data to end users. Also, we evaluated the performance of our protocol under different parameter settings. Therefore, our proposed model provides more security than other models.

6.2 Future Work

Future work in this area mainly includes performance improvements with respect to computation time in the proposed solution. Also, we will try to improve the security. In our model, we suppose that C_1 and C_2 are semi-honest which means that all of them follow the protocol; Therefore, in the future we will extend our protocol to make it secure against malicious protocol. Moreover, we will investigate and extend our research to other complex queries and real setting over encrypted data.

7 References

[1] A. Cavoukian, "Data mining staking a claim on your privacy," in *Computer and Communication Technology (ICCCT), 2012 Third International Conference on*.

[2] Amazon, "What is cloud computing?" 2018. [Online]. Available: https://aws.amazon.com/what-is-cloud-computing/

[3] A. S. J. T. Michael Hogan, Fang Liu, *"NIST Cloud Computing Standards Roadmap"*, 2011.

[4] M. Kuribayashi and H. Tanaka, "Fingerprinting protocol for images based on additive homomorphic property," *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2129–2139, 2005.

[5] M. Bibi, R. Mehboob, S. Shabbir, and S. Khalid, "Data mining in cloud computing applications," 2015.

[6] D. Flair, "Clustering in data mining." [Online]. Available: https://data-flair.training/blogs/cluster-analysis-data-mining/

[7] S. Saxena, "Basic concept of classification (data mining), url = https://www.geeksforgeeks.org/basic-concept-classification-data-mining/."

[8] K. Borne, "Association rule mining – not your typical data science algorithm), url = https://mapr.com/blog/association-rule-mining-not-your-typical-data-science-algorithm/."

[9] B. K. Samanthula, "Privacy-preserving outsourced collaborative frequent itemset mining in the cloud," in *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017, pp. 4827–4829.

[10] O. Goldreich, "The foundations of cryptography, vol. 2. general cryptographic protocols, chap," 2004.

[11] W. K. Wong, D. W. Cheung, E. Hung, B. Kao, and N. Mamoulis, "Security in outsourcing of association rule mining," in *Proceedings of the 33rd international conference on Very large data bases*. VLDB Endowment, 2007, pp. 111–122.

[12] C.-H. Tai, P. S. Yu, and M.-S. Chen, "k-support anonymity based on pseudo taxonomy for outsourcing of frequent itemset mining," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 473–482.

[13] F. Giannotti, L. V. Lakshmanan, A. Monreale, D. Pedreschi, and H. Wang, "Privacy-preserving mining of association rules from outsourced transaction databases," *IEEE Systems Journal*, vol. 7, no. 3, pp. 385–395, 2013.

[14] X. Yi, F.-Y. Rao, E. Bertino, and A. Bouguettaya, "Privacy-preserving association rule mining in cloud computing," in *Proceedings of the 10th ACM symposium on information, computer and communications security*.   ACM, 2015, pp. 439–450.

[15] R. S. Agrawal and R. Srikant, "R. fast algorithms for mining association rules," in *Proceedings of the 20th International Conference on Very Large Data Bases, VLDB*, 1994, pp. 487–499.

[16] X. Yi, R. Paulet, and E. Bertino, *Homomorphic encryption and applications*. Springer, 2014, vol. 3.

[17] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *International Conference on the Theory and Applications of Cryptographic Techniques*.   Springer, 1999, pp. 223–238.

[18] Y. Elmehdwi, B. K. Samanthula, and W. Jiang, "Secure k-nearest neighbor query over encrypted data in outsourced environments," in *2014 IEEE 30th International Conference on Data Engineering*.   IEEE, 2014, pp. 664–675.

[19] B. K. Samanthula, H. Chun, and W. Jiang, "An efficient and probabilistic secure bit-decomposition," in *Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security*.   ACM, 2013, pp. 541–546.