DOI: 10.21917/ijsc.2021.0345

SECURE ASSOCIATION RULE MINING ON VERTICALLY PARTITIONED DATA USING FULLY HOMOMORPHIC ENCRYPTION

M. Yogasini[1] and B.N. Prathibha[2]

[1]*Department of Computer Science and Engineering, Manonmaniam Sundaranar University, India*
[2]*Department of Computer Science, G. Venkataswamy Naidu College, India*

Abstract

Cloud Computing is a leading innovation technology that guides to access applications over the web. The data owner's data can be gotten to and controlled in the cloud. Privacy has become conclusive in information-driven applications within the distributed outsourced information. There are numerous inquiries still coming up on the best way to accomplish a confided in a climate that monitors application and information in the cloud from unapproved users. For offering protection to the users, sporadically there is a need to encode the data before accomplishing any other process. The cryptography method is embraced for data privacy. In this paper, a privacy-preserving construction is intended for vertically partitioned data in the cloud with the assistance of the Fully Homomorphic Encryption method. In this work, the homomorphic encryption and the Fully Homomorphic Encryption method is taken into consideration. The performance of the Rule Mining algorithm namely Eclat is compared with the encryption algorithms. The examination result shows that the Fully Homomorphic Encryption is less time-consuming to generate rule in the cloud, regardless of the number of transactions.

Keywords:

Frequent Itemset Mining, Association Rule Mining, Homomorphic Encryption, Fully Homomorphic Encryption

1. INTRODUCTION

Several Information proprietors can stow data as a joint database in the cloud. Data owners accept that the information put away in the cloud issure with no spillage. All information proprietors will have their exchange segments in the joint database. Data are partitioned as horizontal and vertical in the cloud and the tradesrets and client privacy of a transaction are included. To mine the most esteemed information, privacy-preserving mining is applied to the cloud data. Encryption -based methodology is taken into account in the privacy-preserving data mining atmosphere [1]. Two or a lot of parties mine their information within the cloud with collaboration but nobody is eager to expose their information.

In Transactional information, concealed patterns and relationships are applied to shield the information. Encryption is used to protect the data from third parties. To convert the plain text into ciphertext is termed as encryption, whereas the opposite technique of encryption is known as decryption. In this paper, the computational time of Homomorphic encryption and Fully Homomorphic Encryption are compared. Homomorphic encryption adopts algebraic procedures on the cipher text whereas Fully Homomorphic Encryption return only the encrypted result.

Data mining techniques such as frequent itemset [2] and association rule mining [3] are applied to the transaction data to find the frequent items obtained by the customer. To conceal sensitive items from an unapproved client, the relation between frequent items in a transaction is to be found. This is because to assess customer conduct in business. Frequent Itemset in a transaction is produced by all items-sets whose support ≥minsup [4] [5]. If a set of frequent items have been discovered, then the Association Rules are generated in a transaction.

In this work, the main task is to keep up the security and protection for the information that is transferred and not to allow the unauthorized clients to access the data. Fake transactions are added to the original data in the transaction against frequency analysis attack. Association Rule Mining algorithm namely Eclat is adopted for building Association Rule with different *k*-values, where *k* signifies the items in a transaction. The result depicts that Eclat algorithm for Fully Homomorphic Encryption consume less time for rule generation in cloud when compared to homomorphic encryption.

2. RELATED WORK

Tremendous exploration work had been conveyed in the most recent years. Different procedures had been proposed for the privacy-preserving outsourced data. A portion of the refereed works are ephemeral.

Sandeep and Liji [6] proposed the Advanced Encryption Scheme (AES) Algorithm. This technique scrambles the information before redistributing to avoid the weakness of 'Known Plaintext' assault in the current framework. In this system various information proprietors can do the mining assignments cooperatively. Eclat Algorithm had used to perform mining tasks in the database. Even though the execution time is prominent, it gives better security to the information things and Utilization of assets at the information proprietor's side less. Pallier Encryption method was proposed by Priya et al. [7] to protect the data from unauthorized users. FP-Growth algorithm was adopted by the author and it has perfect execution over Apriori. The result depicts that this method is flexible for a large number of transactions and provides effective security to the data. This framework takes less execution time in terms of privacy concern.

Rozenberg et al. [8] proposed Two-parties and *N*-parties algorithms to discover large itemsets in a vertically distributed transaction. These algorithms were depended on the utilization of fake transactions. Master and Slave technique is implemented. It adopts the *N*-1 slave technique. There is *N*-1 slaves and one Master is participated in a transaction. The Master do the overall calculations and encourage the results to slaves. For each proposed algorithm the author gave and examination of its security and revealed data measure. This method accomplishes great security and protection with less computational cost.

Xun and Tao [9] proposed efficient Fully Homomorphic Encryption for security purpose. Cipher text framework activities in Gentry, Sahai and Waters (GSW) and vector additions in Ducas

and Micciancio (DM) is used to develop the scheme. Error rate of this encryption is low when compared to other two algorithms. The accuracy, security and relevance and present of a transaction is examined with DM and GSW. FHE is suitable for general security safeguarding calculations in reality.

Kenta and Masami [10] proposed asure Association Rule mining organization for vertically partitioned data. Where data owners can impart various records with candidate itemset. The data owner can impart different information or more security and with less calculation cost. Scalar item calculation can be supplanted by this calculation by embracing adaptable data sharing. Effectiveness and security of the information is yet to be developed.

Muthu Lakshmi et al. [11] implemented a model under the concept of Scalar Product with various cryptography techniques like encryption, decryption and scalar product. This model is intended for vertically partitioned data with *n* number of destinations to discover worldwide association rules. This schema gives effective protection to user's data. This technique is not suitable for enormous database.

Chavan et al. [12] have used two frequent itemset mining algorithms such as Dist-Eclat and BigFIM. These algorithms are applied in the Map-Reduced platform which is suitable for online application. These algorithms are applied to predict customer perception and business benefits are maximized at a high level.

Urvashi [13] compared the Eclat algorithm with Apriori Algorithm to find frequent items in a transaction. Apriori algorithm scans the database several times to find frequent items that lead to high computational time, whereas one-time scanning is applied in the Eclat algorithm. So, the computational time of Eclat is less when compared to Apriori. This algorithm is suitable for a large number of transactions.

Lichun Li et al. [14] designed an Efficient Homomorphic Encryption for privacy-preservation of data. Cloud-aided frequent itemset mining was proposed to construct an association in a vertically partitioned data. This solution protects data owner's data from the third parties and from the cloud. Information proprietor can outsource their data in the cloud with a significant level of protection without compromising the performance.

Kaosar et al. [17] adopted two-party association rule mining protocol and the privacy of large database is maintained by implementing fully homomorphic encryption. Support and Confidence of association rules are figured and the result was reimbursed in a single bit.

2.1 METHODOLOGY

In Full Encryption Key Cryptography framework, the data owners have their personal database. Fictious transactions are added to the original data base for privacy concern and are encrypted in the pre-processing stage. The encrypted data are stored as joint database in the cloud to generate Association Rules. Association rules are generated for the frequent items in a transaction. In order to hide the frequent items from unauthorized user, fake transactions are added to the transaction. The cloud restores the encrypted mining results to the information proprietors and they decrypt the outcomes to recuperate the real association rules. The framework for distributed data mining on joint database is portrayed in Fig.1. In Fig.1, AR stands for Association Rule and ARM strands for Association Rule Mining.

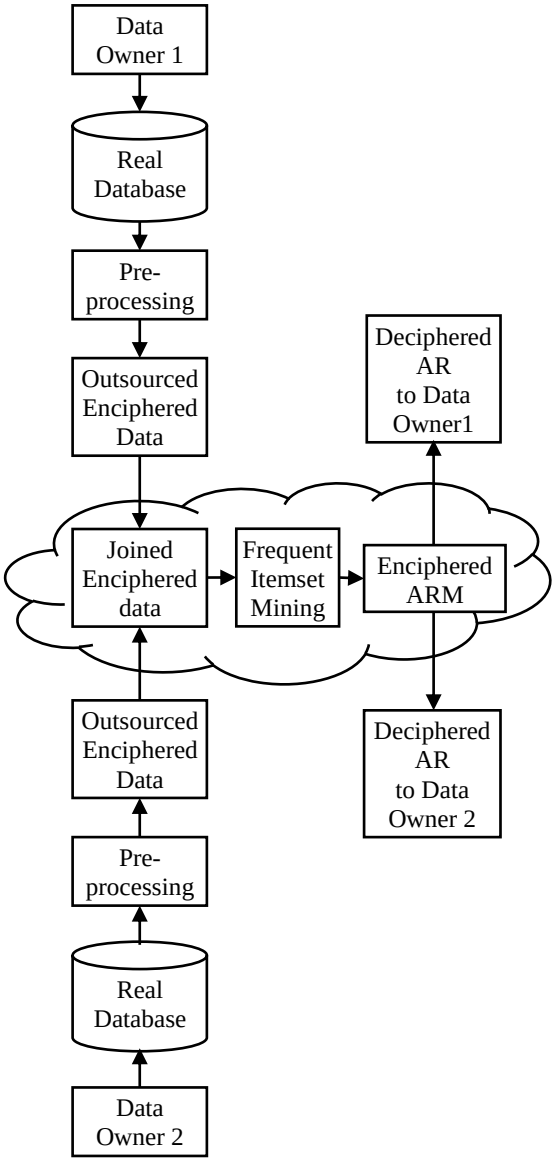

Fig.1. Framework model of outsourced cloud data

2.2 HOMOMORPHIC ENCRYPTION

Homomorphic encryption is utilized for privacy-preserving cloud storage and computation. This permits information to be scrambled and out-sourced to business cloud conditions for handling with encryption. In homomorphic encryption algebraic operations can be done on the cipher text to produce result in cipher text. The decoded text gives similar outcome as acquired when the tasks would be completed on the plaintext. Additive and multiplicative property are the two homomorphic encryption properties. Enc (x+y) and Enc(x*y) is the additive and multiplicative property that can compute from Enc (x) and Enc(y), where the value of x and y are unknown values. The computation "●" is used on the cipher text of m 1 and m2 in order to find the sum of plaintexts in additive homomorphic encryption to find the sum of plaintexts m1+m2 without decrypting m1 and m2.

In multiplicative homomorphic encryption, the computation "⊗" is applied on the cipher text of m1 and m2 in order to find the sum of plaintexts m1+m2 without decrypting m1 and m2. [14]

2.3 FULLY HOMOMORPHIC ENCRYPTION

Fully Homomorphic Encryption (FHE) is discovered in 2009 for security concern of the data. This encryption supports arbitrary computation on ciphertext. This system empowers the development of projects for any needed functionality, which can be track on encoded contributions to create an encryption of the outcome. The encrypted data can be controlled by an untrusted party without uncovering its information sources and inner state. FHE have extraordinary practical ramifications in the distributed of private computations with regards to distributed computing.

The improvement of FHE is a progressive development, incredibly expanding the extent of the calculations which can be applied to measure encoded information homomorphically. With this encryption the cloud can accomplish calculations for the benefit of the client and return just the encoded outcome. Fully homomorphic Encryption (FHE) is equipped for both addition and multiplication quite few times and makessure multi-party calculation more competent. This encryption can deal with subjective calculations on the cipher texts. Enhanced Homomorphic Cryptosystem (EHC), Algebraic Homomorphism Encryption Scheme based on Fermat's Little Theorem (AHEF) are the examples of FHE.

C_1 and C_2 be the two cipher texts

$$C_1 = m_1 e \bmod n$$

$$C_2 = m_2 e \bmod n$$

The Homographic Encryption is

$$C_1 . C_2 = \text{Epk}(m_1).\text{Epk}(m_2)$$

$$= m_1.m_2 \ (\text{mod}n) = (m_1.m_2)(\text{mod}n)$$

$$= \text{Epk}(m_1.m_2)$$

Key Generation, Encryption, Evaluation, Decryption are the operations performed in Homomorphic Encryption. A fully homomorphic encryption scheme is a C-evaluation scheme (Gen, Enc, Eval, Dec) that is compact, Correct and where C is, the set of all circuits. [15].

2.4 ENCODING AND DECODING ALGORITHM

Gen $(1\beta,\gamma)\gamma)$ is the significant key production calculation. This algorithm receipts two data sources, safety constraint β and supplementary input γ, and yields a tripartite key (ak,bk,evk), where ak is the key utilized for encipher, bk is the key utilized for unscrambling together with the key evk is the key utilized for assessment.

$Ec(ak,x)$ is the encipher procedure. While it takes the input as encipher key ak and an unencrypted manuscript x. Crypto text c is the output of the calculation.

$Evl(evk,C,c_1,\ldots,c_n)$ is the assessment calculation. This procedure considers assessment key evk as a data source, an orbit $c \in C$ along with a triplet of sources of info possibly a blend of cryptogram together with past assessment outcomes which yield an estimation productivity.

$Dc(bk,c)$ is the unscrambling calculation. Decipher key bk is consider as input and constructs the unencrypted manuscript x.

$$\text{Gen}: N \times A \rightarrow kp \times ks \times ke$$

$$Ec: kq \times P \rightarrow x$$

$$Dc: kr \times \text{Z} \rightarrow P$$

$$Evl: kt \times \text{C} \times W^* \rightarrow Y$$

kq, kr, kt are the respective key spaces of pk, sk and evk. The public key encompasses and explanation of the unencrypted manuscript and cryptogram position. The contribution to the key origination procedure Gen can be represented as a single symbolization.

Personal Information ought not to be quickly apparent. It expresses that the yield from the assessment of a particular procedure on code manuscripts resembles the outcome from the encipher of an unencrypted manuscript esteem x, produced for this situation by the calculation and the comparing plaintexts. The encoded information construct appropriations thar are measurably adjacent, either algebraically near, or indistinguishable.

2.5 FREQUENT ITEMSET MINING

Frequent itemset is a basic part of numerous Data Mining assignments that attempt to discover fascinating patterns from databases. Frequent Itemset Mining is a technique used for market basket analysis. The uniformities of consumers in supermarkets, companies are to be found by adopting frequent itemset mining and is used to find set of products that are often purchased together. In a transaction minimum support values are fixed by the data owner and it denoted as Lk, where k is the size of the itemset. An itemset is considered as A frequent itemset whose support is larger than the user specified minimum support value. Frequent Items in a transaction is covered by every information proprietor by embedding fictitious transactions to the database.

Similar recurrence with k-1 items is shared by every item in the similar database. This procedure is done in the wake of infixing the fake exchange. If an itemset in a transaction have support ≥minsup, then that item is known as frequent item. From the joint database, frequent items can be generated by the cloud by comparing the item set's support value with the support threshold value. If Suppcount(I)≥T_s, the data owner decrypt I's Encrypted Support Verifying Result (ESVR) [16] to determine whether "I" is a real frequent item or not. The information proprietor decodes the ESVR of every frequent itemset to hit whether the itemset is actually frequent or not. The information proprietor can decrypt the actual frequent item in a transaction.

2.6 ASSOCIATION RULE MINING

Association Rule Mining is a standard based strategy to discover relations between things in a transaction and to conjecture the sign of a thing dependent on the various things in the exchange. In vertical database, on the basis of frequent itemset cloud aided Association Rule Mining solution was formed. $X \rightarrow Y$ is the representation of Association Rule, where X and Y are two items-sets in a transaction. An antecedent (if) and a consequent (then) are the two parts of the Association Rules. An item found in data is termed as antecedent and an item found in blend with the antecedent is termed as Consequent. The relationship of items in a transaction is obtained by the two parameters namely support and confidence. The frequent occurrence of the if/then association in a transaction is known as support and confidence states that the number of times these occurrences found to be true.

Association Rule Candidate $X \rightarrow Y$ that satisfies $X \cap Y = \emptyset$ and X∪Y, where X, Y and X∪Y are seemingly frequent item-sets. If $Conf(X \rightarrow Y) \geq Tc$, the cloud brings back the association rule candidates, ESVR to data owners. The values are decrypted by the information proprietor to invent the association rules. The data owner decrypts the ESVR of $X \cup Y$ for a candidate $X \rightarrow Y$. This was taken over to find whether $Supp(X \cup Y) \geq T_s$ or not. If the result is positive, the ECVR of $X \rightarrow Y$ will be decrypted by the data owner in order to find whether $Conf(X \rightarrow Y) \geq T_c$ or not. The items X and Y are decrypted to regain the real association rule in the plain text only when the result is true. This is done by the data owner. The fake transactions are cancelled by the use of ERV in the mining results. For the same plain texts, the ERVs are different. Association rule candidates, ESVR and ECVR are decrypted by the data owners for the purpose finding association rules [14].

3. EXPERIMENTAL ANALYSIS

The retail and the pumsb repositories are taken into account for the evaluation. These datasets are an open-source dataset. The computation cloud time of four dissimilar transactions with diverse k-values is evaluated. Retail dataset is a market basket data of 88,162 transitions. Pumsb is a census data of population and housing and encompasses 49,046 transactions. The performance of the éclat algorithm is evaluated independently for the different k values in the transaction set with the help of Python. The transaction set taken for the trial was 5000, 10000, 15000, 20000 with different k-values.

Table.1. Computation Time with different support value for Pumsb Dataset

Transaction	5000		10000		15000		20000	
Min Support	Eclat Homo	Eclat FHE	Eclat Homo	Eclat FHE	Eclat Homo	Eclat FHE	Eclat Homo	Eclat FHE
0.3	5.1	4.6	6.5	5.4	7.5	6.5	8.5	7.1
0.5	4.5	3.5	5.4	4.2	6.2	5.4	7.6	6.2
0.7	3.8	2.7	4.3	3.4	5.7	4.6	6.2	5.4
0.9	2.6	1.8	3.4	2.3	4.8	3.8	5.5	4.6

The Table.1 represents the cloud execution time for different transactions with the minimum support value of 0.3,0.5,0.7,0.9 of pumsb dataset. The outcome demonstrates that the proposed FHE method produces less computation time. For minimum support value 0.7 with 15000 transactions, the yield attained in Eclat FHE bring down 19% of outsourced computation time.

Table 2. Computation Time with different support value for Retail Dataset

Transaction	5000		10000		15000		20000	
Min Support	Eclat Homo	Eclat FHE	Eclat Homo	Eclat FHE	Eclat Homo	Eclat FHE	Eclat Homo	Eclat FHE
0.3	6.2	5.5	7.1	6.6	8.5	7.2	9.2	8.2
0.5	5.4	4.2	6.4	5.7	7.4	6.2	8.5	7.4
0.7	4.5	3.4	5.2	4.9	6.3	5.7	7.2	6.5
0.9	3.4	2.4	4.5	3.7	5.8	4.9	6.4	5.2

The Table.2 illustrates the outsourced implementation time for the retail database. With the minimum support value 0.5, the computation time for Eclat FHE is 13% lesser than Eclat Homo method. From Table.1 and Table.2, it is spotted that the performance time deviates with improved support values.

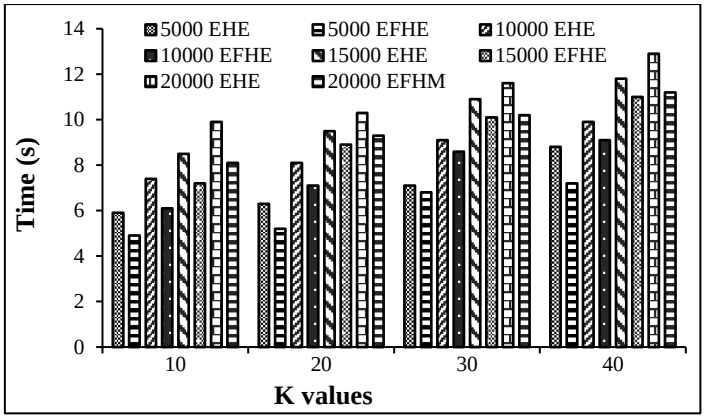

Fig.2. Execution Time with different k-values for Pumsb dataset

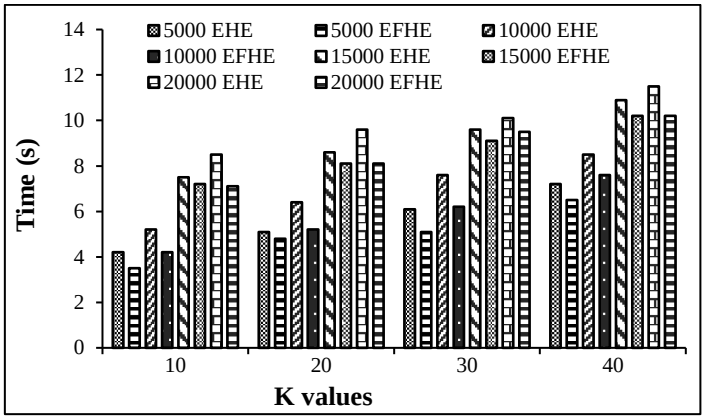

Fig.3. Execution Time with different k-values for Retail dataset.

The Fig.2 represents the outsourced cloud time for different transactions like 5000, 10000, 15000, 20000. Running time for different k-values of the different transactions are calculated and compared. It is detected that cloud execution time of the Eclat for FHE is less when compared to Homomorphic Encryption even the size of the transaction is increased.

From Fig.2 and Fig.3, it is pragmatic that the running time differs with increasing values of k. The retail dataset's execution time is less when compared to pumsb data set for different k vales. For pumsb dataset the running time for FHE is 3.19s, 4.57s, 5.641 and 6.12s lesser than Homomorphic Encryption for 5000, 10000, 15000 and 20000 transactions respectively with different k-values. Similarly, the execution time of FHE is 3.01s, 4.47, 5.127 and 5.965s lesser than HE for different transactions. In Fig.2 and Fig.3, EHE, EFHE stands for Eclat Homomorphic Encryption and Eclat Fully Homomorphic Encryption correspondingly.

4. CONCLUSION

In this paper, privacy-preserving system for vertically partitioned outsourced data using Association Rule Mining is

embraced. This framework encourages the information proprietors to re-appropriate their information as a joint database with appropriate frequent itemset mining. Privacy-preserving Association Rule Mining with the assistance of the Eclat algorithm was developed to mine the information and precisely furnish the outcome with less calculation time.

This algorithm keeps up stable security and execution of Rule mining when the number of transactions is expanded with various k-values. The computation time of FHE is less when compared to Homomorphic Encryption for different transactions since the FHE produce cipher text with an irregular clamour to guarantee semantic security. From the examination, it is discovered that this FHE framework is proficient and encourages the information proprietors to outsource their information with an elevated level of protection without negotiating the performance.

REFERENCES

[1] V. Sathya and Dr. V. Gayathri, "Encryption-Based Techniques for Privacy Preserving Data Mining", *International Journal of Scientific and Engineering Research*, Vol. 8, No. 4, pp. 52-56, 2017.

[2] M. Kamber and J. Pei, "*Mining Frequent Patterns, Associations and Correlations*", Morgan Kaufmaan Series, 2012.

[3] M. Kaur and S. Kang, "Market Basket Analysis: Identify the Changing Trends of Market Data using Association Rule Mining", *Procedia Computer Science*, Vol. 20, No. 1, pp.78-85, 2016.

[4] A.M. Shahiri, W. Hussain and N.A. Rashid," A Review on Predicting Student's Performance using Data Mining Techniques", *Proceedings of 3rd International Conference on Information Systems*, pp.414-422, 2015.

[5] S.S Shengzhi0 and X. Cheng, "Differentially Private Frequent Itemset Mining via Transaction Splitting", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 27, No. 7, pp.1875-1891, 2015.

[6] Sandeep and Liji, "Secure Outsourced Association Rule Mining using Homomorphic Encryption", *International Journal of Engineering Research and Science*, Vol. 3, No. 9, pp. 1-13, 2017.

[7] R. Priya and Shweta, "A Tow Way Encryption for Privacy Preservation of Outsourced Transaction Database for Association Rule Mining", *International Journal of Scientific Research in Science and Technology*, Vol. 4, No. 5, pp. 276-285, 2018.

[8] Boris and Ehud, "Association Rules Mining in Vertically Partitioned Databases", *Data and Knowledge Engineering*, Vol. 59, No. 2, pp. 378-396, 2006.

[9] Xun and Tao, "More Efficient Fully Homomorphic Encryption Scheme Based on GSW and DM Scheme", *Security and Communication Networks*, Vol. 2018, pp. 1-14, 2018.

[10] Kenta and Masami, "Secure Association Rule Mining on Vertically Partitioned Data Using Private-set Intersection", *IEEE Access*, Vol. 8, pp. 1-10, 2017.

[11] N.V. Muthu Lakshmi and K. Sandhya Rani, "Privacy Preserving Association Rule Mining in Vertically Partitioned Databases", *International Journal of Computer Applications*, Vol-39, pp.29-39, 2012.

[12] K. Chavan, P. Kulkarni and P. Ghodekar, "Frequent Itemset Mining for Big Data", *Proceedings of International Conference on Green Computing and Internet of Things*, Vol. 1, pp. 1365-1368, 2015.

[13] Urvashi Garg, "Eclat Algorithm for Frequent Itemsets Generation", *International Journal of Computer Systems*, Vol. 1, No. 3, pp. 82-84, 2014.

[14] L. Li and R. Lu, "Privacy-Preserving Outsourced Association Rule Mining on Vertically Partitioned Database", *IEEE Transactions on Information Forensics and Security*, Vol. 11, pp. 1847-1861, 2016.

[15] Imran and Archana, "Homomorphic Encryption Applied to Cloud Computing", *International Journal of Information and Computation Technology*, Vol. 4, No. 15, pp. 1519-1530, 2014.

[16] S. Varma and P.I Liji, "Secure Outsourced Association Rule Mining using Homomorphic Encryption", *International Journal on Engineering Research*, Vol. 3, No. 9, pp.70-76, 2017.

[17] G. Kaosa, R. Paulet and X. Yi, "Secure Two-Party Association Rule Mining", *Proceedings of Australasian Conference on Security*, pp.15-22, 2011.