SECURING HEALTH CARE DATA IN COLLABORATIVE DATA PUBLISHING USING MAPREDUCE FRAMEWORK

Shital S. Suryawanshi[1], Vinod S. Wadne[2]

[1] *PG Student, Computer Engineering Department, Savitribai Phule Pune University, JSPM's Imperial College of Engg. & Research, Wagholi, Pune, India.*

[2] *Assistant Professor, Computer Engineering Department, Savitribai Phule Pune University, JSPM's Imperial College of Engg.& Research, Wagholi, Pune, India.*

Abstract - *Today publishing data on a web becomes need. Publishing such microdata can breach privacy of any individual. For analysis purpose researcher, medical practitioner required such health related data. Existing system used encryption algorithms for securing data. The data is stored on HDFS by encrypting and the user having key can only access that data by decrypting it. Big data is heterogeneous, distributed data where data is collected from different sources having different dimensions. In hospital patients data can be stored in different form such as audio, video, and in images. Big data having different characteristics like variety of data, its volume and velocity, this makes it different from other databases. Data privacy is one of the challenge in data mining with big data. Big data keeps growing continuously. In case of big data it is not efficient to encrypt large amount of data as it is time consuming. Existing provider aware algorithm has problem of data loss due to insider attack. K-anonymity and l-diversity are very popular algorithms for generalization and bucketization. They have some their own little limitations. In insider attack provider can infer the information of other user using his own records and with some background knowledge. To preserving the privacy of the user we need to use some method so that data privacy is preserve and at the same time increase the data utility. In the proposed system we focus to maintain the privacy for distributed data, and overcome the problems of M-privacy using new updated provider algorithm with a slicing technique. The main goal of paper is to publish an anonymized view of integrated data, which will be immune to attacks. We also use MR-Cube method which is used to compute large cube with non algebraic measures such as TOP-k, count.*

Key Words: *MapReduce, MR-Cube, Data privacy, Slicing, Data anonymization*

1. INTRODUCTION

Big data contains variety of data such as audio, video, images, text and having large volume in terms of size, velocity (from batch processing to real time processing). The data is collected from multiple sources and having different dimensions [1]. Big Data is generally indicating petabyte or terabyte data, with some characteristics which makes it different from other databases. Conventional tools such as relational databases are failed to handle, process, manage, and analyze data. To explore the large volumes of data and extract useful information for future actions is the fundamental challenge for big data applications. Big data is in structured, unstructured and semi-structured format. Using traditional tools it is difficult to solve problem related with big data. It's become challenge to mine knowledgeable information from large dataset for future use. There are different challenges of Data mining with Big Data. One of them is data privacy challenge, which can be solved using different approaches like key based encryption [3] and anonymization. To process and compute high dimensional distributed data is one of the challenge. In this the data has different dimensions for e.g. in hospitals, patients data is stored in text and images , videos are used to stored results of X-ray, CT scan for detail examinations. MR-Cube approach is used for efficient computation of cube [7].

Performing data analysis on big data becomes expensive due to its nature and the data is distributed, continuously keeps growing. For analyzing multidimensional data, data cube is powerful tool. Consider a data warehouse maintain the sales information containing <city, country, state, day, month, year and sales>. Where city, country and state attribute are of local dimension and attribute day, month, and year are of temporal dimension. Cube analysis provides convenient way to discover insight from the data by computing aggregate measures. Top-Down approach, Bottom-Up Computation (BUC), A Mining Cubing Approach, Parallel approach [14] is some of the cube computation techniques. There are two main limitations in the existing cube computation techniques. First they are design for single machine or cluster with small number of nodes. It is difficult for businesses or companies containing huge data storage. Second limitation is many technique use algebraic measure to avoid processing groups with a large number of tuples. There is need of technique to compute large cube efficiently in parallel. MapReduce programming paradigm is used to analyze and process such large scale data.

Existing encryption algorithms stores data on HDFS by using encryption technique and therefore the access to retrieve data becomes limited. As the big data is distributed large volume of data it's becomes tedious job

to every time encrypt data as the data grows continuously. The analyzers required data for analysis to take some decisions. Previous work shows anonymization algorithms such as centralized algorithms and distributed algorithms also have some drawbacks to provide privacy to high dimensional data. Centralized algorithm [8] assumes that, to processed data, the data should be located at one location it means all data should fit in memory which is not possible for large dataset. Even if we have that much memory to process large amount of data the moving cost of data is high. The distributed algorithms mainly focus at security integrating and anonymizing multiple data sources rather than scalability issue. The proposed systems focus on maintaining the privacy for distributed data, and also overcome the problems of M-privacy using updated provider aware algorithm.

As there is increasing need of sharing personal information from distributed database, the special care should be taken to protect it from attacker. Attacker can be single entity or group of entities. In m-privacy multiple providers want share their data securely. In m privacy case one or more data owner (provider) will be willing to know the records or data of other provider. The attacker can breach privacy with the help of his own record and with some background knowledge. Collaborative data publishing considered as a multi-party computation problem. In collaborative data publishing providers want to compute an anonymized view of their data without disclosing any private and sensitive information. A data recipient that might be an attacker, e.g., q0, attempts to gather additional information about data records using the published data, D*, and background knowledge, BK. For example, k-anonymity [9] protects against identity disclosure attacks by requiring each quasi identifier equivalence group (QI group) to contain at least k records. L-Diversity [20] requires that each QI group should contain at least l "well-represented" sensitive values. Differential privacy guarantees that the presence of a record cannot be inferred from a statistical data release with little assumptions on an attacker's background knowledge.

We considered a potential attack on collaborative data publishing. For anonymization k-anonymity and l-diversity algorithm used but they have problem of data loss and identity disclosure problem. In existing provider aware algorithm at least k * 2 – 2 number of record goes in waiting forever. In proposed system using new updated provider aware algorithm data utility get increases. We used slicing algorithm for anonymization and verify it for security and privacy by using provider aware algorithm of data privacy. Our main goal is to publish an anonymized view of integrated data, which will be immune to attacks. We also use MR-Cube technique to compute large cube. Using MR-Cube data extraction and data insertion time get improve and more accuracy will get in result. We improve the security and privacy with the help of slicing technique which fulfils privacy verification with better performance than provider aware (base algorithm) and encryption algorithm.

2. RELATED WORK

Big data is generated and collected from various autonomous, heterogeneous sources and data having different dimensions [1]. The Big data are continuously growing is become the challenge to processing large data and securing that data. The different characteristics of big data makes difficult to process, manage, and stored it securely. Ingale et a1. [3] Proposed Advance Encryption Standard (AES) with k-anonymization for privacy conserving to achieve privacy. K-anonymization allows database to maintain a suppressed and generalized form of data. A different anonymization algorithm with different operations has been proposed [8] [9] for privacy preservation. Today the dataset becomes very large for example sensor data, web data, data on social networking sites, anonymizing such data becomes challenge for traditional anonymization techniques. To analyze this large amount of data various cube computation techniques [14] have been used. The existing cube computation techniques have several limitations that they compute cube over limited number of node and many techniques compute with algebraic measure only. Nandi et al. [5] proposed MR-Cube approach for efficient cube computation with holistic measure for large dataset.

Fung extended the k-anonymization algorithm to preserve the information for cluster analysis. The major challenge in this is the lack of class labels that could be used to guide the anonymization process. The solution is to first partition the original data into clusters on the original data then problem is converted into counterpart problem for classification analysis where class label encode the cluster information in the data and then apply TDS to preserve k-anonymity.

FUNG et al. [8] proposed a new privacy model LKC-privacy to overcome the challenges of traditional anonymization methods using centralized and distributed anonymization algorithm. A data structured TIPS (Taxonomy Indexed partitionS) is exploited in centralized algorithm to improve efficiency of TDS. molloy at el. [2] used slicing, which partitions the data both horizontally and vertically. Slicing preserves better data utility than generalization and can be used for membership disclosure protection. Another important advantage of slicing is that it can handle high-dimensional data. Slicing can be used for attribute disclosure protection. Generalization and bucketization are anonymization techniques. For generalization k-anonymity algorithm is very popular and l-diversity is used for bucketization. In both of these approaches the attributes are partition into three types. The first is identifier like ID No or SSN, second is Quasi-identifier which is combination of more than one attribute and the third is sensitive attribute. For anonymous data these identifiers are first remove form data and then partition into bucket.

Identity disclosure, Attribute disclosure and membership disclosure are threat in privacy preservation, which needs to overcome. Slicing is used on high dimensional data

prevent membership disclosure reduce the information loss and increased data utility. Top down specialization approach [17] used to anonymized large scale data set on cloud. Existing TDS approaches for large scale data sets having scalability problem. The centralized TDS approaches use TIPS to improve the scalability and efficiency by indexing anonymous data records. Centralized approaches suffer from low scalability and efficiency when it handles large scale data sets. The assumption in centralized approach to fit all data in memory for processing which is not possible for large data sets. To handle scalability issue we used MR-Cube approach which first generate annotated lattice and then used it to perform main MR-Cube MapReduce.

3. PROPOSED SYSTEM

3.1 Problem Definition

While maintaining the data privacy in big data or in a multi-provider environment various challenges are faced, like data security, data utility and data processing. Existing provider aware algorithm was not allowing enough data utility while securing data. In the proposed system we focus to maintain the privacy for distributed data, without loss of data and also overcome the problem of M-privacy using new updated provider aware algorithm.

Our main goal is to publish an anonymized view of integrated data, which will be immune to attacks. We improve the security and privacy with the help of slicing technique which fulfils privacy verification with better performance than provider aware (base algorithm) and encryption algorithm.

This Proposed work having lot of enhanced techniques to preserve the privacy in collaborative data publish. Thus the all techniques will preserve the membership disclosure and provide more data utility than the existing system. Basically slicing is the important algorithm with all available methodologies like data publication, bucketization and generalization in the proposed system. With slicing we used MR-Cube method, combining this two methods we get more accurate data with security and privacy.

In collaborative data publishing with data partitioned across multiple data providers, each contributing a subset of records di. A data provider could be the data owner itself who is contributing its own records. Requirement is to publish an anonymized view of the integrated data such that a data recipient including the data providers will not be able to compromise the privacy of the individual records provided by other parties, considering different types of malicious users and information they can use in attacks. In Privacy for collaborative data publishing, focus is on insider attack by colluding data provider who can use their own data record to understand the data records shared by other data record providers. This problem can be resolved by using different approaches such as m-privacy, Heuristic algorithms, Data provider aware anonymization Algorithm and SMC/TTP protocols.

M-Privacy [18] protects anonymized data against m-adversary (is a situation where data providers are using combination of data for breaching the anonymized records) with respect to given privacy constraint. M-Privacy can also be guaranteed when there are duplicate records; it also includes syntactic privacy constraint, differential privacy constraint and monotonicity of privacy constraints. M-privacy verification: Binary m-Privacy verification algorithm, Top-Down and Bottom-Up algorithms are used for this. This verification process first analyze the problem by modeling adversary space and using heuristic algorithms with effective pruning strategy and adaptive ordering techniques for effectively checking m-privacy with respect to equivalence group monotonicity constraints.

3.2 Architecture of Proposed System

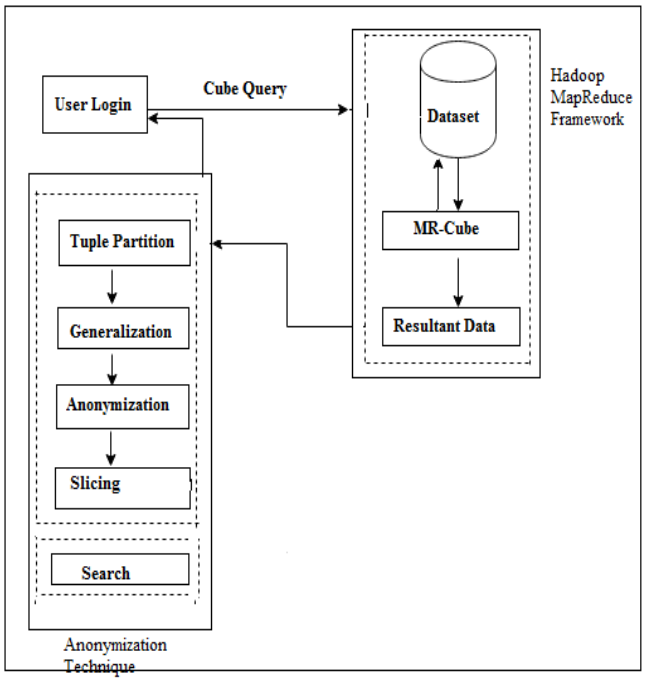

Fig -1: Proposed System Architecture

In Fig.1 shows the architecture of proposed system which contains user module, MR-Cube module, Anonymization modules. In first module the user can be an administrator, authorized user or providers which already have an account. In our project administrator can view all the data of doctors, patients, and also all providers. He can add disease. Administrator has all the permissions to view, add, and remove any data. In provider login particular authorized provider can only view the data related to his patients. Provider can add the patient's information. Doctors also can view the data or records of his patients only.

We can do either search or view anonymized data. We can search patients with particular disease group by provider or zip code using MR-Cube and anonymization technique. The resultant data is anonymized (T*) and count is shown. We can also view all slicing process where data is sliced then bucketized. In this process we used updated provider aware algorithm so that only the records which satisfy constraint are displayed and remaining records stored in new bucket. This algorithm again applies on the other bucket and the record which goes in bucket just because they are not satisfying the constraint not because they breaching privacy are display in final bucket.
Here we are going to use Hadoop framework for managing large scale distributed data and processing it. User passes a query as input in Hadoop framework where data node is master node which receives this query. Cube query is generated annotated lattice which is further given to process main MR-Cube MapReduce Process. The cube query generate materialized data which is distributed to mappers. The final output of MR-Cube is given as input to anonymized technique. As we describe above the anonymized technique shows more secure and accurate result of records.

3.3 Description of the Terms and Proposed Algorithm

3.3.1 Anonymization

Slicing is basically depends on Attribute and tuple partitioning. In Attribute partitioning (vertical partition) we partitioned data as name, age-zip and Disease and tuple partitioning (horizontal partition) as t1, t2, t3, t4, t5, t6. In attribute partitioning age and zip are partitioned together because they both are highly correlated because they are quasi identifiers (QI). These QI can be known to attacker. While tuple partitioning system should check L diversity for the sensitive attribute (SA) column. Algorithm runs are as follows.
Step1. Initialize bucket k=n, int i= rowcount, column count=C, Q=D, // D= data into database, Arraylist= a[i];
Step2. While Q is not empty If i≤n Check L diversity; Else i++; Return D*;
Step3. Q = Q -(D* + a[i]);
Step4. Repeat step 2 and 3 with next tuple in Q
Step5. D* =D * U A[D] // next anonymized view of data D

First initialize k = limit of data anonymization bucket size, number of rows, number of columns, array list and database in the queue(step 1). Further process will done if and only if queue is not empty i.e there should be data in database. Check data for L diversity if rowcount = k = m (step 2). Initially Q= Queue of data. If our bucket data fulfill k anonymity and L diversity, it return D* i.eanonymized view of data. The data from the database which cannot fulfill requirement of privacy will stored in arraylist a[i]. Now data remains in database i.e in Q=Q-D*+a[i] (step3). Repeat step 2 and step 3. A[D] is anonymization of data in database. Apply above steps for remaining data and create new anonymization view which is the union of original view and new one i.e D*=D*UA[D].

3.3.2 L-diversity

Ldiversity is the concept of maintaining uniqueness within data. In this system we used this concept on SA (Sensitive Attribute) i.e on disease. Our anonymized bucket size is 6 and I maintain L=4 i.e from 6 disease record 4 must be unique.

Step1. Initialize L=m, int i;

Step2. If i= n-m+1; Then a[0]..a[1], insert these values as they are in Q; i++;

Step3. Else Check privacy constraint for every incremented value in Q If L=n then Fscore=1 Insert value in the row i++; else Add element to arraylist a[i];

Step4. Exit

First initialize L=m and rowcount i. If i=n-m+1 i.e if k=n=6 and L=m=4 then i=3, upto third row data doesn't need to check for Fscore. Add this data as they are coming from Q(step 1 and 2). For further data from Q check data for privacy constraint. If data fulfills L , then Fscore=1. If data doesnt fulfill Fscore=1, then add element in array list a[i] (step 3).

3.3.3 Permutation

Permutation means rearrangement of records of data. Permutation process is used for re-arrangement of quasi identifier i.e Zip-Age.

3.3.4 Fscore

Fscore is privacy fitness score i.e the level of fulfillment of privacy constraint C. If fscore=1 then C(D*)= true.

3.3.5 Constraint C

C is a privacy constraint in which D* should fulfill slicing condition with L diversity as explain above. Consider value of L diversity is 4. Fscore should be 1 when system fulfills L diversity condition.

3.3.6 Some verification processes are carried out are

3.3.6.1 Verification for L diversity

For verification of L diversity I used Fitness score function. For checking L diversity generate continuous similar values of SA i.e insert similar disease. Check for Fscore=1. If L=m, return Fscore. If privacy breachi.e if anonymized view take data as insertad then it breached privacy. D* shoud take data which fulfill L= m.

1. Generate continuous similar values of SA

2. Check for privacy constaint and fscore=1;

3. If Privacy breach; Then early stop; Else Return (Fscore);

4. Exit

3.3.6.2 Verification for strength of system against number of provider

For verification against number of provider, add one more attribute in anonymized data as a provider to output. This verification will prove that our technique of anonymization doesn't depend on number of provider. Existing system i.e provider aware anonymization algorithm depends on database as well as provider.

1. Generate values of SA by providers P= 1..n

2. Check for privacy constraint and Fscore=1 with respect to number of provider

3. If Privacy breach; Then stop; Else Return (Fscore);

4. Exit

3.3.6.3 Provider aware algorithm for reduce the time complexity

Input: Data set with D, providers n, with C

Output: Slice view (T*) with provider

1. read data from (D up to null)

2. for each (attributes in table) for each (tupels in tables)

3. Set quasi identifier (QIfr) and sensitive attributes (SA)

4. Apply generalization technique it will classify the tuples in QIfr groups

5. Apply anonymization on relative information attributes

6. While(verify data-privacy(D, n, C) = 0) do if (Di→D) verified with QIfr then add Di up to when K-anonimity else early stop Bucket(i1)→D;

7. permute the data with (I=(I(null-1)))

8. Apply Pruning on (D)

9. Apply step 1, 2, 3 on Bucket (i1)

10. if (C fails with (D)&&(p#1)) Bucket(i2)→Bucket(i1(j))

11. Display all (Bucket (i2)6=null)

12. end while

13. end for

3.3.8 MR-Cube

MR-Cube approach used to efficiently compute large cube. It addresses the challenges of large scale cube computation with holistic measures. The complexity of cubing task is depending on data size and cube lattice size.

4. MATHEMATICAL MODEL OF PROPOSED SYSTEM

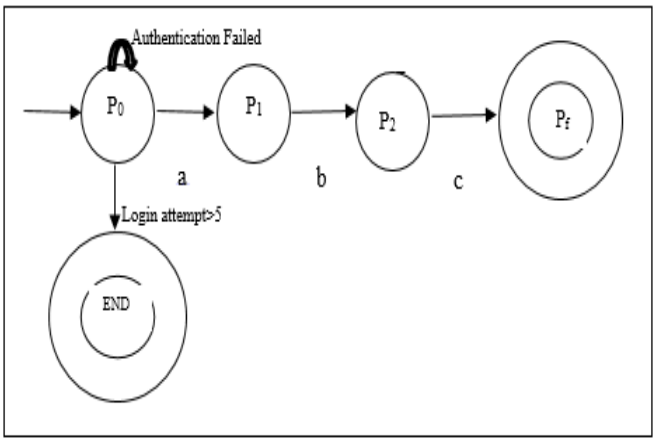

Fig -2: DFA of Proposed System

DFA= {Q, ∑, δ, q0, F}
Where
Q=Finite Set of States
∑=Input Alphabet
δ=Transition between states
P0=Initial State
F=Final State

Q={P0, P1, P2, Pf}
P0=Initial State
P1= Create Cube Query
P2=MapReduce
Pf=Anonymization

∑= {a, b, c} Where
a=Query with parameter

b=Materialized Data
c=Resultant MapReduce Data

Table -1: State Transition Table

	a	b	C
P0	P1	Φ	Φ
P1	Φ	P2	Φ
P2	Φ	Φ	Pf
Pf	Φ	Φ	Φ

5. EXPERIMENTAL RESULTS

We present here sets of experimental results to 1) compare and evaluate query processing time in Hive and by generating cube 2) evaluate and compare proposed updated provider aware algorithm for given dataset to get more data utility with secured data.

5.1 Experimental Setup

In this section our goal is to evaluate the proposed algorithm that is Updated Provider Aware Algorithm in terms of utilizing more data utility and MR-Cube Approach for efficient cube computation so the data extraction time get reduced. We used healthcare dataset which contains different attribute like name, age, zip, address, disease etc. 1 Lacks of records have been used in all experiment. The Disease has been used as a sensitive attribute (SA). This attribute has 10 distinct values. Data are distributed among 4 providers' p1,p2,p3,p4. The privacy constraint C is defined by k-anonymity and l-diversity. C is conjunction of both k-anonymity and l-diversity. Anonymization use Fscore i.e. privacy fitness score, if the diversity is 3 the fitness score is 1, for diversity 5 the Fscore will 2. All experiment were conducted on Intel Pentium 1.60 GHz PC with 4 Gbyte RAM and 60.2 GB Hard disk.

A. Query Processing with cube

We used Hadoop Apache open source framework for storing and processing data. Hive is used as a sql in Hadoop. We generate a patient cube with four dimensions (name, disease, age, doctor) and two measures (provider and zip). We compare time required to processed query in existing system in Hive and using MR-Cube. In fig.3 data extraction performance is shown. Here the comparison shows by building a cube lattice over the dataset we can retrieve data in less time with more accuracy than time required to data retrieving through Hive.

Fig. 4 shows the performance of data insertion in proposed system. In existing encryption based system required large amount of time to insert data as it used very lengthy process. Fig.5 shows the time required to slicing in existing encryption based system and in proposed system.

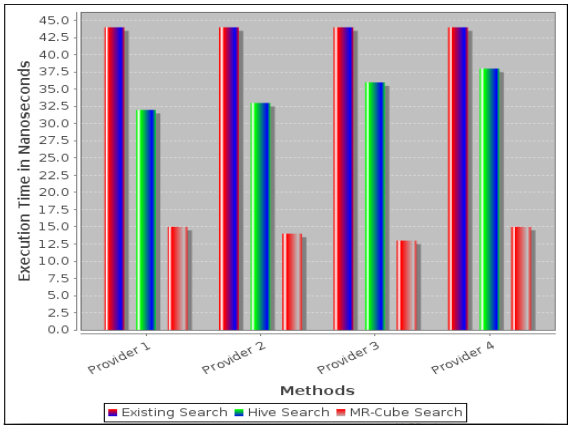

Chart -1: Data Extraction Performance

B. Provider Aware Algorithm for Increasing Data utility

We used Updated provider Aware Algorithm in proposed system to reduce time complexity and it efficiently utilized data as compare to existing algorithm. In existing system only the data witch satisfy the constraint can consider secure to display, but the data which is not anonymous but just because it does not satisfying constraint it goes in waiting forever. Here the data get loss. In proposed system we are keeping that all data in bucket and applying the technique on that data, again the remaining data goes in final bucket and the final bucket will display securely.

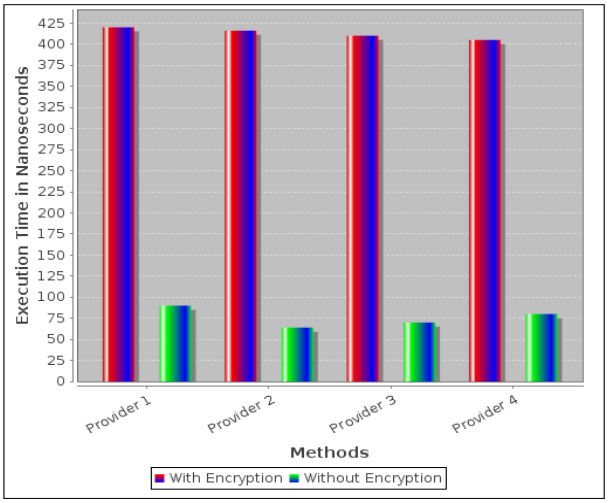

Chart -2: Data Insertion Performance

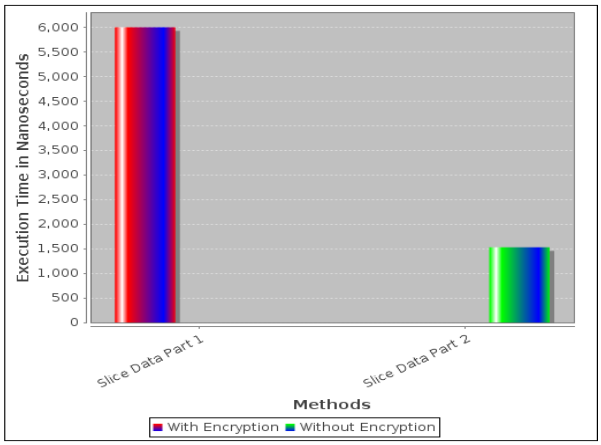

Chart -3: Slicing Performance

6. CONCLUSIONS

The result shows that the proposed Updated Provider Aware Algorithm performs better than existing one. The more data get utilized for analysis without breaching privacy of individual. Existing system uses encryption before inserting/ storing data and decryption using key to retrieve it. It required large amount of time and it was not worked efficiently for big data. Using MR-Cube approach data cube can compute efficiently. The cube has been generated for given dataset with dimensions and measures. MR-Cube compute cube with holistic measures like Top-k query so get accuracy. The Provider Aware algorithm reduces time complexity as existing system uses multiple checks for privacy constraint.

ACKNOWLEDGEMENT

I would like to thanks to my Professors and colleagues for their guidance and helped me to expand my horizons of thought and expression. I would also like to give special thanks to my family members to encourage, support and for giving their valuable times.

REFERENCES

[1] Xindong Wu, Fellow, IEEE, Xingquan Zhu, Senior Member, IEEE, Gong-Qing Wu, and Wei Ding, Senior Member, IEEE " Data Mining with Big Data" in IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 1, JANUARY 2014

[2] Tiancheng Li, Ninghui Li, Jian Zhang, Ian molloy "Slicing: A New Approach for Privacy Preserving Data Publishing" in IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 24, NO. 3, MARCH 2012

[3] Madhuri Patil, Sandip Ingale "Privacy Control Methods for Anonymous & Confidential Database Using Advance Encryption Standard" in International Journal of Computer Science and Mobile Computing, Vol. 2, Issue. 8, August 2013.

[4] Senthil Raja M & Vidya Bharathi D "Enhancement of Privacy Preservation in Slicing Approach Using Identity Disclosure Protection" in ITSI Transactions on Electrical and Electronics Engineering (ITSI-TEEE) Volume -1, Issue -2, 2013.

[5] Arnab Nandi, Cong Yu, Phil Bohannon, Raghu Ramakrishnan "Distributed Cube Materialization on Holistic Measures"

[6] Zhengkui Wang, Yan Chu, Kian-Lee Tan, Divyakant Agrawal, Amr EI Abbadi, Xiaolong Xu, "Scalable Data Cube Analysis over Big Data" appliarXiv:1311.5663v1 [cs.DB] 22 Nov 2013

[7] Arnab Nandi, Cong Yu, Philip Bohannon, and Raghu Ramakrishnan, Fellow, IEEE, "Data Cube Materialization and Mining overMapReduce" TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 6, NO. 1, JANUARY 2012

[8] NOMAN MOHAMMED and BENJAMIN C. M. FUNG, PATRICK C. K. HUNG, CHEUK-KWONG LEE, "Centralized and Distributed Anonymization for High-Dimensional Healthcare Data" in ACM Transactions on Knowledge Discovery from Data, Vol. 4, No. 4, Article 18, Pub.date: October 2010

[9] C.Aggarwal, "On K-Anonymity & the Cure of Dimensionality" Proc. Int'l Conf. Very Large data Bases (VLDB), PP, 901-909, 2005
[10] Benjamin C.M. Fung, Ke Wang, and Philip S. Yu, Fellow, IEEE, "Anonymizing Classification Data for Privacy Preservation" in EEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 19, NO. 5, MAY 2007
[11] D. Mohanapriya, Dr.T.Meyyappan, "High Dimensional Data Handling Technique Using Overlapping Slicing Method for Privacy Preservation" in International Journal of Advanced Research in Computer Science and Software Engineering Volume 3, Issue 6, June 2013
[12] Machanavajjhala and J.P. Reiter, "Big Privacy: Protecting Confidentiality in Big Data," ACM Crossroads, vol. 19, no. 1, pp. 2023, 2012.
[13] K. V. Shvachko and A.C. Murthy, "Scaling Hadoop to 4000 Nodes at Yahoo" Yahoo! Developer Network Blog, 2008.
[14] Dhanshri S. Lad, Rasika P. Saste, "Different Cube Computation Approaches: Survey Paper" (IJCSIT) International Journal of Computer Science and Technologies, Vol. 5 (3), 2014, 4057- 4061.
[15] Hadoop. http://hadoop.apache.org/.
[16] The Apache Software Foundation http://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/YARN.html
[17] Xuyun Zhang, Laurence T. Yang, Senior Member, IEEE, Chang Liu, and Jinjun Chen, Member, IEEE "A Scalable Two-Phase Top-Down Specialization Approach for Data Anonymization Using MapReduce onCloud" in IEEE Transaction on Parallel and Distributed System, vol. 25, No. 2 February 2014
[18] Slawomir Goryczk Li Xiong Emory, Benjamin C. M. Fung, "m-Privacy for Collaborative Data Publishing"
[19] Prashanth Mohan, Abhradeep Thakurta, Elaine Shi, Dawn Song, David E. Culler, "GUPT: Privacy Preserving Data Analysis Made Easy" in SIGMOD '12, May 20–24, 2012
[20] Ashwin Machanavajjhala, Johannes Gehrke, Danial Kifer "ℓ-Diversity: Privacy Beyond k-Anonymity.