International Journal of Engineering Sciences Paradigms and Researches [Volume 47, Issue: Special), March 2018]
www.ijesonline.com ISSN (Online): 2319-6564

FP-Growth Algorithm-Based Big Data Security Development in Frequent Itemset

Sushree Sheetal Beura
Raajdhani Engineering College, Bhubaneswar
sushreebeura@rec.ac.in

Abstract *- Big data is employed in many different contexts these days. The data is processed before being saved and includes a variety of information, including sensitive information. It is also safeguarded by privacy security measures. Meaningful information is provided for the stored data, and those who can access it are guaranteed security. The vast collection of data was gathered from human behavior and interactions. According to a security perspective, the largest group of items—that is, the most data—are appropriately analyzed and user information is safeguarded. A rise in big data volume can also lead to an improvement in user privacy, which can improve data utility, time efficiency, and privacy level.*

Key Words**:** Big data, Frequent Item set, Data Privacy, Data Security, Apriori algorithm, FP-Growth algorithm.

1. INTRODUCTION

In technology development, the amount of big data is generated by social networking like sensor network, health care applications and other company through the internet. The simulating situation of data includes data storage, data analysis, sharing, transfer, querying, data privacy and security. The data generation is growing tedious and rapid development and it is very difficult to handle it using traditional method. Based on this definition, the properties of big data are volume, velocity and variety. The volume denotes the amount of data to be generated.

Fig 1: Different Types Of Data In Big Data

The new data are generated and characterized as velocity. This multifariousness of data is referred as variety. The variety of data as video, audio, image and even text. Veracity represents the accuracy and confidence of data.

1.1 LIFE CYCLE OF BIG DATA

The data life cycle is the sequence of stages that can run a specific unit of data, that unit of data go through from its initial generation to its continuous archival or deletion at the end of its useful life cycle of big data. In any other life cycle of data are it can be created, shared, maintained, archived, retained and deleted. In this case, the data is not necessarily made available to the public but is just sent outside to the business operation or organization. If the data is no longer processed or accessed by the user it stored and can access the data whatever we need for future. It consists of data generation, data processing and data storage. The end goal of any big data is to produce an effective data product.

Data generation: Data can be generated from various disturbed sources like large, diverse and complex. It is awkward for manage traditional system in colligate with proper domain such as internet, business and research. Data storage: It involves storing and managing large amounts of data sets. This system consists of two class that are hardware infrastructure and data management. Hardware infrastructure refers to utilizing information and communication technology resources for various task. Data management refers to set of software deployed on hardware infrastructure to manage and query large data sets. Data processing: it refers the process of data collection, data transmission, pre-processing and retrieving useful information.

Fig 2: Big Data Life Cycle

1.2 PRIVACY AND SECURITY ENVIRONMENT IN BIG DATA: SECURITY FOR BIG DATA

Authentication: It is the process to determine whether the identity of users, services and hosts are whom they claim to be. Authorization: It is the process to determine which permission a person, data, service and system. It can be seen

as both the preliminary setting of permission by a system administrator and the actual checking of the permission values when a user obtains access. Data protection: Ensure that only authorized users have access to accurate and complete information when required. The main goal is to guarantee data is appropriately protected from modification or disclosure.

PRIVACY FOR BIG DATA: DATA PROTECTION TECHNOLOGY

If the data with privacy confidential, the attacker can't incur the effective value of data. And also, we can use Data Encryption Technology, Data Anonymity Technology, Generalization Technology, So the privacy protection technique generalizes the original data, the value of the original data address has become not clearly understood, then we achieve the intent of privacy protection.

Access Control Technology: In big data environment, the number of users is vast, the authority is complex, and a new technology adopted to sharing the data using the Access Control Technology. It consists of Role mining technique as Role Based Access Control (RBAC) and Risk Adaptive Access Control (RAAC). To achieve a risk-based access control to define, determine and quantify the risk of the big data environment. Data Provence Technology: It is necessary to record the origin and the process of calculation and provide additional support for the mining and decision. The established security mechanism to protect data can be divided into four categories. They are file level data security, database level data security, media level security, and application level encryption security. The method of data province is labelled, through this label, we can identify the data in table which is source and it can access easily, checking the correctness of result, or update the data with the minimum cost.

2. RELATED WORK

J. Manykiya, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh and A. Byers, "Big data: the next frontier for innovation, competition, and productivity", in that, the data generation rate is growing so rapidly that it is becoming extremely difficult to handle it using traditional methods[1].

J. Gantz and D. Reinsel, "Extracting value from chaos", in this, the big data is defied as a new generation of technologies and architectures, designed to economically extract value from very large volumes of a wide variety of data, by enabling high velocity capture, discovery and analysis based on the volume, velocity and variety [2].

H. Hu. Y. Wen, T.S. Chua, and X. Li, "Toward scalable systems for big data analytics: A technology tutorial", in this, life of big data in data storage using hardware infrastructure and data management methods [3].

Gang Zeng, "Big data and information security", in this, data privacy protection technology and data security technology are using access control technology and data provence technology [4].

Sagar Bhise, Prof. Sweta kale, "Efficient algorithms to find frequent itemset using data mining", in this, frequent itemset mining algorithm incur the high degree of privacy, data utility, and high time efficiency. The private frequent pattern growth algorithm is divided into two categories as pre-processing and mining phase. The algorithm for implementing mining sequence item sets as Apriori Algorithm and FP growth algorithm are used. The pre-processing phase consists to improve utility, privacy and splitting method and the mining phase consists to transaction splitting and run time estimation to find given item set [5].

3. EXISTING SYSTEM

The PFP growth algorithm is categorized as pre-processing and mining phases. The pre-processing phase consists to improve utility, privacy and smart novel splitting methods to transform the data into the database. It is performed only once. The Apriori and FP- growth algorithm are most common used one. In Apriori algorithm is a breath first search algorithm. And it scans a database with the maximal length of the frequent item sets which has value as one. For an better quality of apriori algorithm which needs to scan the input data items at only once. The apriori algorithm used only for frequent item set but FP-growth algorithm used data intensive and also computing intensive. The mining phase consist to information lost during the transaction splitting and calculates a run- time estimation methods to find actual item set in given database. And the advanced stage of dynamic reduction method is used to dynamically reduce the noise and to guarantee privacy during the mining process of item set.

Based on this noise, first estimate the actual support in transformed database and compute actual support to the original database. And in mining phase, estimate the frequent item set of threshold value based on maximal support.

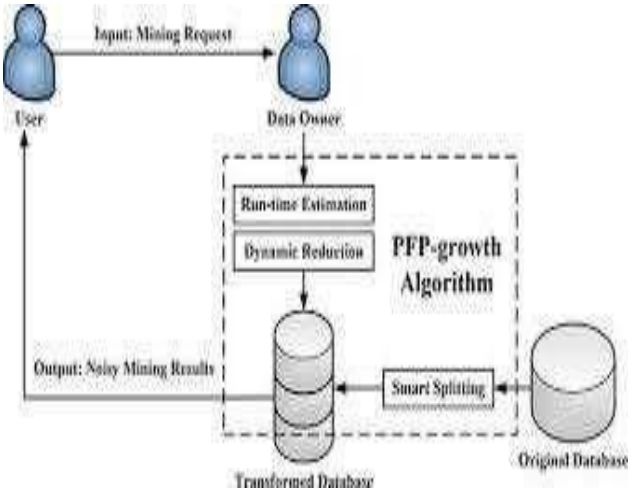

Fig 3: Block Diagram of PFP-Growth Algorithm

4. PROPOSED SYSTEM

4.1 FREQUENT PATTERN MINING

The item mining is most important problem in the data mining. The future prospect of this design, to protect a user data as private in frequent item set mining algorithm to receive high degree of privacy, data utility, and high time efficiency.
It is an efficient and scalable method for mining the complete set of frequent patterns by pattern fragment growth, using an extended prefix-tree structure for storing compressed and crucial information about frequent patterns named frequent-pattern tree (FP-tree).

4.2 FP-GROWTH ALGORITHM

It is an alternative way to find frequent item sets without using candidate generations, thus improving performance. This algorithm works as follows:

1. First it compresses the input database creating an FP-tree object to represent frequent item set. And then it divides the compressed database into a set of conditional
2. Databases, each one associated with one frequent pattern.
3. Finally, each such database is mined separately.
4. Using this strategy approach, the FP-Growth item set reduce the search costs looking for short patterns recursively and then concatenating them, so it provides the long frequent patterns.

The possible to find the complete set of frequent patterns item sets using the FP-growth algorithm are:

Input: constructed FP-tree
Output: complete set of frequent patterns
Method: Call FP-growth (FP-tree, null).
Procedure FP-growth (Tree, α)
{
1) if Tree contains a single path P then α with support = minimum support∪
2) for each combination do generate pattern β of nodes in β.
3) Else For each header ai in the header of Tree do { α with support = ai.support;∪
4) Generate pattern β = ai
5) Construct β.s conditional pattern base and then β.s conditional FP-tree Tree β
6) If Tree β = null
7) Then call FP-growth (Tree β, β)}
}

4.3 FP-TREE STRUCTURE

The frequent pattern tree is a compact structure that stores quantitative of large information about frequent patterns in a database when we used. It is further shown that parallel approach is much more scalable than a serial implementation for one machine.

1. One root labelled as "null" with a set of item-prefix sub trees as children, and a frequent-item-header table.
2. Each node in the item-prefix consists of three fields as Item-name: registers which item is represented by the node, Count: the number of transactions represented by the portion of the path reaching the node, Node-link: links to the next node in the FP tree carrying the same item-name, or null if there is none.
3. Each entry in the frequent item-header table consists of two fields: Item-name: as the same to the node, Head of node-link: a pointer to the first node in the FP-tree carrying the item name.

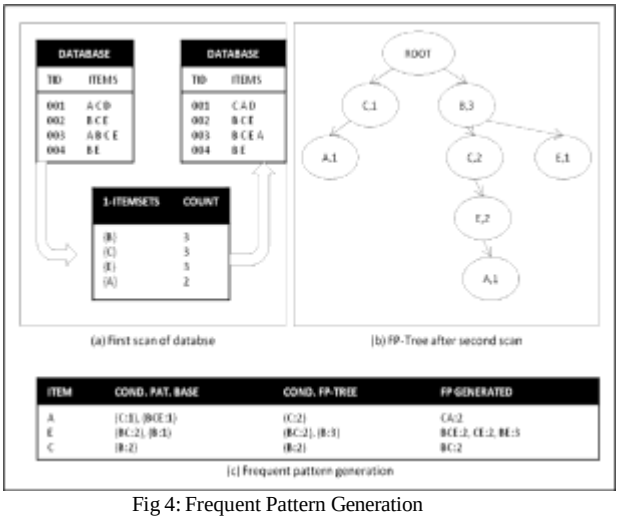

Fig 4: Frequent Pattern Generation

5. RESULTS AND DISCUSSION

The main focus of this work is to study FP-growth algorithm, it divides and compress the input data base and that is associated with any one of the frequent patterns. And each frequent pattern is mined separately. Finally, it concatenates the frequent pattern for a better result. The PFP growth algorithm is time efficient and better utility and good privacy protection. The FP Growth reduces the search costs when we use any other algorithm.

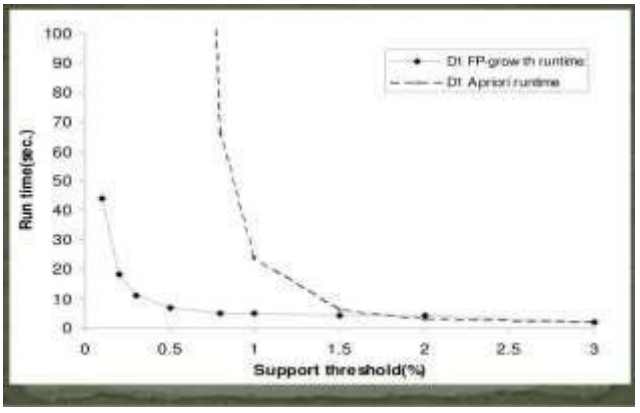

Fig 5: Comparison Of Apriori And FP-Growth at Runtime Using Threshold Value

6. CONCLUSION AND FUTURE WORK

In large database, it is not possible to hold the FP-tree in main memory. May be, in future work, to solve this problem is to firstly partition the database into a set of smaller database and then construct an FP-tree from each of these smaller databases. Regarding this, it may increase the privacy protection technique, data utility, and the time-efficient.

REFERENCES

[1] J. Manykiya, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh and A. Byers, "Big data: the next frontier for innovation, competition, and productivity", Mickensy global institute, pp.1-137, Jun 2011.

[2] J. Gantz and D. Reinsel, "Extracting value from chaos", in Proc. On IDC view, Jun 2011, pp. 1-12.

[3] H. Hu. Y. Wen, T.S. Chua, and X. Li, "Toward scalable systems for big data analytics: A technology tutorial", IEEE Access, vol. 2, pp. 652-687, Jul, 2014.

[4] Gang Zeng, "Big data and information security", IJCER, Vol. 2, Issue. 6, Jun, 2015.

[5] Sagar Bhise, Prof. Sweta kale, "Efficient algorithms to find frequent itemset using data mining", IRJET, Vol. 4, Issue 6, Jun, 2017.

[6] Big data and analytics for variety, volume and velocity http://vusumuzi.dbsdataprojects.com/2017/03/04/big-data-and-analytics.

[7] Frequency pattern item set data diagram in https://www.researchgate.net/figure/FP-Growth-example_fig1_267959080 for FP growth tree structure.

[8] Feng Gui, Yunlong Ma, Feng Zhang, Min Liu, Fei Li, Weiming Shen, Hua Bai, "A Distributed Frequent Itemset Mining Algorithm Based on Spark" Proceedings of the 2015 IEEE 19th International Conference on Computer Supported Cooperative Work in Design (CSCWD).

[9] Hongjian Qiu, Yihua Huand, Rong Gu, Chunfeng Yuan, "YAFIM: A Parallel Frequent Itemset Mining Algorithm with Spark", 2014 IEEE 28th International Parallel and Distributed Processing Symposium Workshops.