ISSN No. 0976-5697

Volume 6, No. 1, Jan-Feb 2015

International Journal of Advanced Research in Computer Science

REVIEW ARTICLE

Available Online at www.ijarcs.info

A Review on Big Data Mining, Distributed Programming Frameworks and Privacy Preserving Data Mining Techniques

Y. Sowmya
Asst. Prof, St. Marys Engineering College
Deshmukhi village, Nalgonda dist
Telangana, India

Dr. M Naga Ratna
Asst. Prof, Dept of computer science and engineering
JNTU College of engineering hyderabad,
Telangana, India

Dr.C Shoba Bindu
Asst. Prof, Dept of computer science and engineering
JNTU College of engineering Anantapur
Andhra pradesh, India

Abstract: Data mining gradually became big data mining as the enterprises are causing exponential growth of data. Comprehensive mining of such data can bestow accurate business intelligence. Towards this end big data mining has become a new buzz word in the mining paradigm. The emergence of technologies such as virtualization and cloud computing paved way for the processing of big data which is characterized by Volume, Velocity and Variety. For big data processing, a new programming model, MapReduce is used. This framework runs in distributed environment to process huge amount of data. There are many distributed programming frameworks such as Hadoop, Haloop, Dryad, Sailfish, and AROM that are based on MapReduce and equivalent programming paradigms. The success of enterprises in future depends on the intelligent mining of big data for comprehensive business intelligence. At the same time privacy preserving data mining also important as the data mining should not be taken place at the cost of privacy. In this paper we explore big data mining, distributed programming frameworks and the privacy preserving data mining practices and techniques.

Keywords: Big data mining, privacy preserving data mining, distributed programming frameworks

I. INTRODUCTION

Big data is the data which has characteristics such as Volume, Velocity and Variety (V3). As the data of enterprises is growing exponentially every year, it became imperative to take care of mining or extracting trends from the huge amount of data in order to make expert decisions from the derived intelligence. In this context, traditional data mining is inadequate and there is need for new algorithms for managing big data. Big data is measured in peta bytes (See Figure 1). Such huge amount of data when processed can provide comprehensive business intelligence. Thus the big data mining needs huge amount of computing resources as well. For this reason it can be done using cloud computing, the new computing model, which enables the storage and processing of huge amount of data.

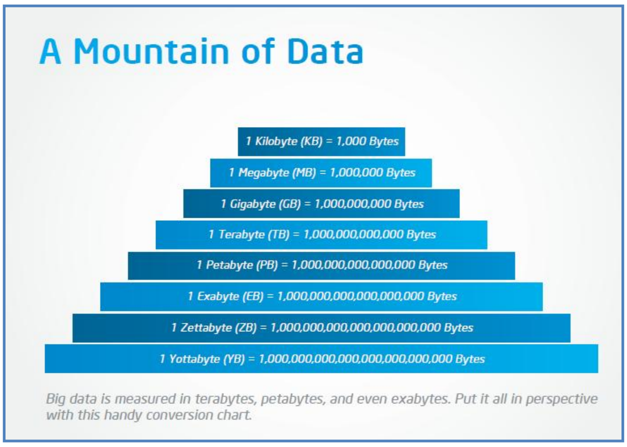

A Mountain of Data
1 Kilobyte (KB) = 1,000 Bytes
1 Megabyte (MB) = 1,000,000 Bytes
1 Gigabyte (GB) = 1,000,000,000 Bytes
1 Terabyte (TB) = 1,000,000,000,000 Bytes
1 Petabyte (PB) = 1,000,000,000,000,000 Bytes
1 Exabyte (EB) = 1,000,000,000,000,000,000 Bytes
1 Zettabyte (ZB) = 1,000,000,000,000,000,000,000 Bytes
1 Yottabyte (YB) = 1,000,000,000,000,000,000,000,000 Bytes
Big data is measured in terabytes, petabytes, and even exabytes. Put it all in perspective with this handy conversion chart.

Figure 1 – Illustrates how big data is measured [32]

Processing huge amount of data can also provide big value to enterprises as they can make well informed decisions that can help them to grow faster. There are many distributed programming frameworks such as Hadoop, Haloop, Dryad and so on that make use of a new programming model named “MapReduce” which can process huge amount of data in distributed environment.

Having said this the data mining can be performed with ease using cloud computing and the traditional data mining techniques are not efficient in the distributed environment where parallel processing can leverage efficiency of IT systems. In this context, it is possible that data mining operations might disclose privacy of individuals. Thus privacy preserving data mining (PPDM) became imperative. Towards this end many techniques like k-Anonymity, l-diversity, and t-closeness came into existence. In this paper we throw light into the big data mining and its challenges, distributed programming frameworks and PPDM techniques. The remainder of this paper is structured as follows. Section 2 provides insights into big data mining. Section 3 discusses various distributed programming framework and their merits and demerits. Section 4 focuses on the PPDM techniques for big data mining. Section 5 concludes the paper besides providing recommendations for future work.

II. BIG DATA MINING

Jones et al. [1] discussed various for large radio arrays with respect to big data challenges. Smith et al. [2] focused on the privacy issues of big data mining associated with social media. They analyzed threat to individuals and their privacy over social networking. Location based big data handling is designed and analysis was made. In process they could analyze the capabilities of social media to protect privacy of users. Begoli and Horey [3] focused on the design principles that can help in mining from big data effectively. The design principles they devised include support for various analysis methods, use different architectures or sizes and make data accessible. Kaisler et al. [4] discussed about issues with big data. They include transport and storage issues, management issues, and processing issues. They also focused on analytical challenges such as machine learning techniques, visualization, video analysis, and cloud computing, data mining algorithms, and analyzing structured data.

Kumar et al. [5] focused on handling big data using cloud computing. They also proposed a secure mechanism to handle data. Wu et al. [6] proposed a theorem known as HACE that can represent and characterize big data. They proposed a framework for big data mining.

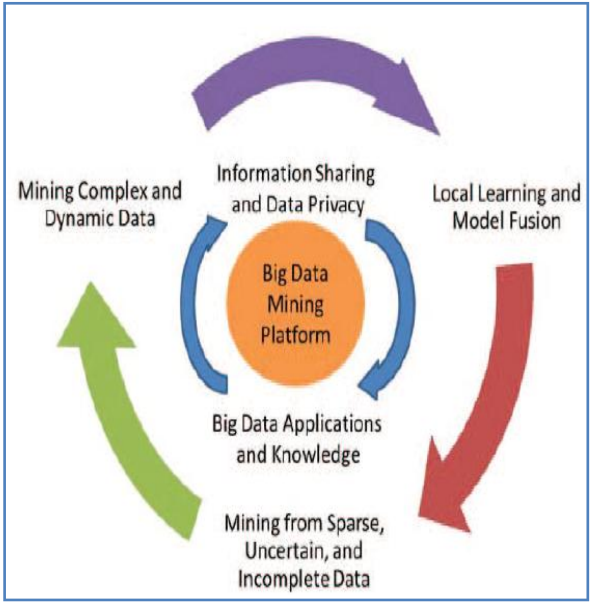


Figure 2 – Framework for processing big data

As can be seen in Figure 2, it is evident that the big data mining platform can a series of activities that can form a life cycle. The operations also include information sharing and data privacy besides big data applications and extracted knowledge that provides comprehensive business intelligence [6].

Yang and Fong [7] focused on the decision tree algorithm named iOVFDT which was used to handle concept-drift problem in big data analysis. Tien [8] made projections on big data mining and its future prospect. Lee [9] also discussed about the future of the organizations in the wake of cloud computing and big data mining. Especially they discussed about "Bring Your Own Device" concept. Katal et al. [10] focused on good practices of big data besides its challenges and issues. They opined that big data mining is required to handle data of IT industries, sensor data, risk analysis, social data, and government data. The challenges and issues include privacy and security, data access and sharing, processing and storage issues, analytical challenges, technical challenges, skill requirement, quality of data, and scalability. The tools and techniques available include Hadoop, MapReduce and distributed programming frameworks. Bertino [11] has thrown light into big data opportunities and challenges. The challenges include data acquisition, information extraction, data integration, query processing, interpretation while there are plethora of advantages such as comprehensive business intelligence for well informed decision making.

Doshi et al. [12] explored the combined usage of SQL and NOSQL approaches for big data mining. They explored the big data mining using ORM techniques such as Hibernate. Chen et al. [13] explored cloud based SpiderMine which is meant for mining big data using cloud computing paradigm. Especially they studied efficient large graph pattern mining. The experimental results revealed that SpiderMine can efficiently mine top-k large patterns from big data. Xxx focused on risk involved in big data mining. The risks include information disclosure and over sharing, risk related to location based information, information aggregation risk, and privacy preserving data mining (PPDM).

III. DISTRIBUTED PROGRAMMING FRAMEWORKS

There are many distributed programming frameworks that allow processing of voluminous data in distributed environment. One of the famous frameworks is Apache Hadoop which makes use of MapReduce, a new model of programming, to process huge amount of data. Figure 3 presents MapReduce operations with Apache Hadoop.

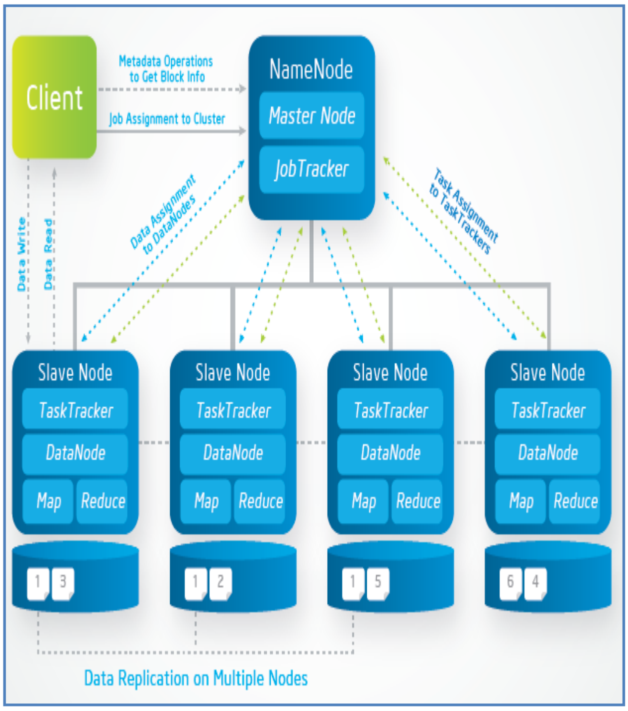


Figure 3 – Architectural overview of Apache Hadoop

As can be seen in Figure 3, it is evident that the architecture shows multiple nodes that act as slave and they work under a master node. There is option pertaining to task assignment and job tracking. This is facilitated in master node while the slave nodes are meant for processing data and give results back to the master. All these things are taken place in direct response to the end users' requirements. MapReduce programming is essential to work with big data as it can leverage the parallel processing of the world. The framework is as shown in Figure 4.

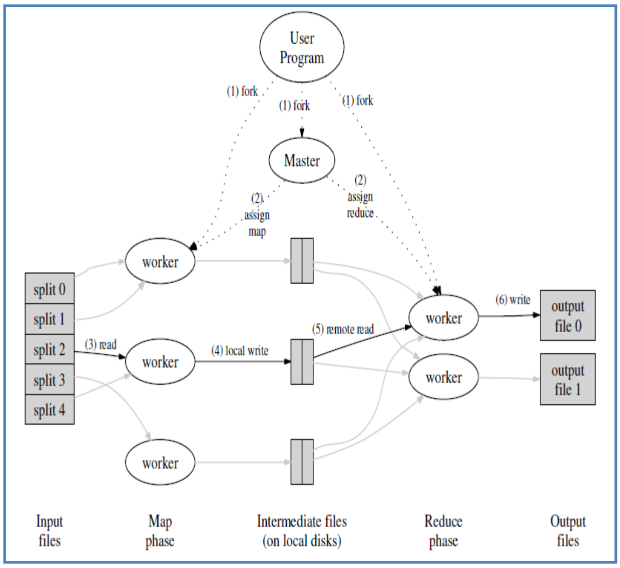


Figure 4 – Map Reduce operations

As can be seen in Figure 4, it is evident that the MapReduce programming has various things such as input files, Map phase which is carried out by worker nodes, intermediate files on local disks and Reduce phase is also carried out by worker node and finally output files are generated and used.

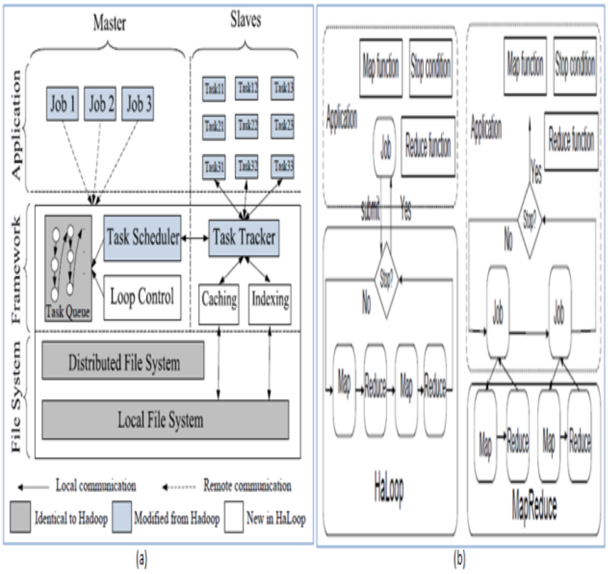


Figure 5 – Haloop compared with Hadoop

Bu, Howe, and Ernst [33] proposed a modified variant of Hadoop that can process data iteratively on large clusters. It extends MapReduce and also provides various capabilities to it including caching mechanisms, loop aware task scheduler and other features. As real time applications need to process huge amount of data in terms of data mining and data analysis Hadoop came into existence. In Hadoop the main programming model is known as MapReduce which is suitable for processing big data. Hadoop is a distributed file system that supports processing huge amount of data in terabytes or more in distributed environments such as cloud computing. As MapReduce is already a scalable and efficient programming model that is improved further. Another tool that has been focused is dryad which is also a popular platform for processing big data.

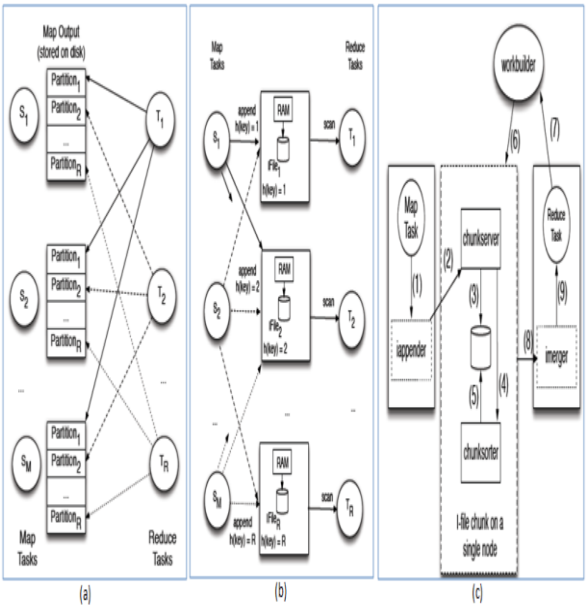


Figure 6 – Sailfish processing huge amount of data

Rao, Ramakrishnan, and Silberstien [34] presented a new framework for processing big data known a Sailfish. It also uses MapReduce layer. However, its design is improved to enhance Map and Reduce phases of the new programming paradigm suited for big data processing. They have built sailfish in such way that it can improve 20% faster performance when compared with Hadoop besides supporting a new future known as auto-tuning. They studied many frameworks and presented Sailfish. The frameworks they studied include MapReduce, Dryad, Hadoop, and Hive. The authors have improved the functionality of MapReduce in their framework by name Sailfish [34]. The map tasks and reduces tasks and their functionality has been improved as shown in figure 7.

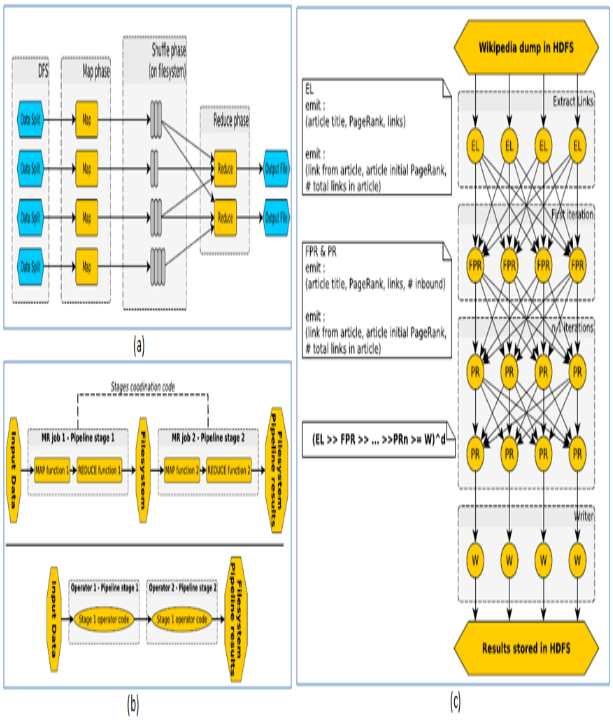

DFS
Map phase
Shuffle phase (on filesystem)
Reduce phase
Data Split
Map
Reduce
Output File
(a)
Stages coordination code
Input Data
MR job 1 - Pipeline stage 1
MAP function 1
REDUCE function 1
Filesystem
MR job 2 - Pipeline stage 2
MAP function 2
REDUCE function 2
Filesystem Pipeline results
Input Data
Operator 1 - Pipeline stage 1
Stage 1 operator code
Operator 2 - Pipeline stage 2
Stage 1 operator code
Filesystem Pipeline results
(b)
Wikipedia dump in HDFS
EL
emit :
(article title, PageRank, links)
emit :
(link from article, article initial PageRank, # total links in article)
Extract Links
EL
FPR & PR
emit :
(article title, PageRank, links, # inbound)
emit :
(link from article, article initial PageRank, # total links in article)
First iteration
FPR
n/i iterations
PR
(EL >> FPR >> ... >>PRn >= W)^d
Writer
W
Results stored in HDFS
(c)

Figure 7 – MapReduce model (a), comparison with Dryad and PageRank of AROM

As can be seen in figure 8 (a), the MapRedue model ha many phases involved. Important phases are Map phase and Reduce phase. First of all, the DFS component take bit data and splits the data. Such data is mapped in the Map phase. Afterwards, the maps are processed using Shuffle phase on the file system. Afterwards, the Reduce phase generates final output. The MapReduce model has some drawbacks. They include mandated shuffle phase is not efficient, and joins are also cumbersome. The other programming model is known as DFG. Based example for DFG is Microsoft's Dryad. The pipelining in Dryad is better than that of MapReduce [35].

IV. PRIVACY PRESERVING KNOWLEDGE DISCOVERY

Privacy Preserving Data Mining (PPDM) has been around for some years which deals with preserving privacy while exposing data for data mining. Verykios et al. [16] classified privacy preserving techniques into data distribution, data modification, data mining algorithm, data or rule hiding, and privacy preservation. The first one refers to the distribution of data, the second one changes made to data, the third one refers to the procedure followed for mining, the fourth one refers to the hiding of rules or data while the fifth one refers to the act of not revealing identity of individuals in the dataset. Wu et al. [17] classified the PPDM techniques with simplified classification.

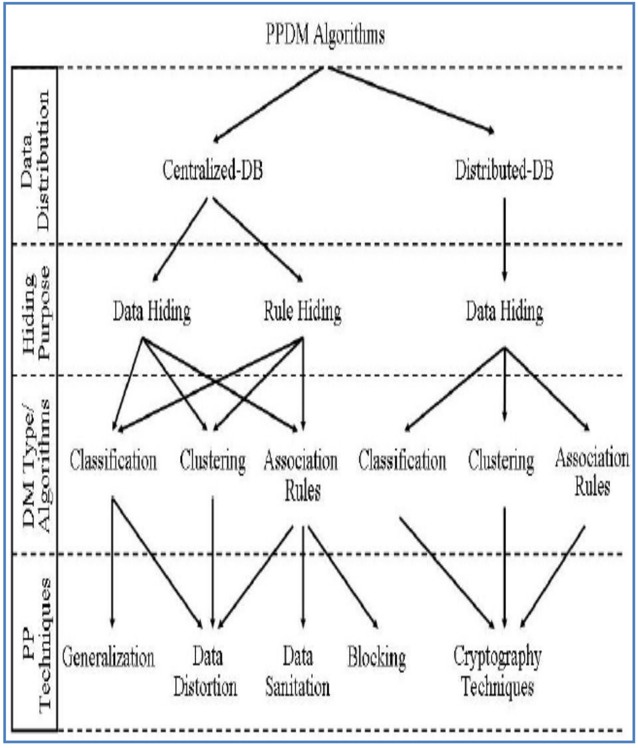
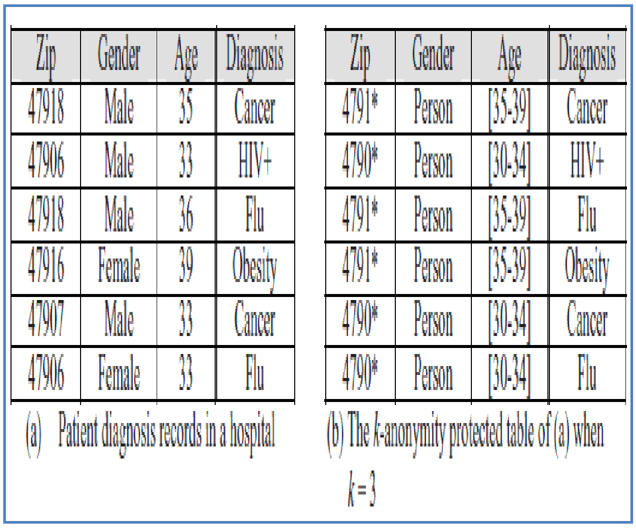


Figure 1 – Summary of PPDM algorithms

As can be seen in Figure 1, the PPDM techniques were classified into the techniques related to data distribution, hiding purpose, DM types or algorithms and PP techniques [17]. Chiu and Tsai [18] used k-anonymity model for privacy preserving data mining. Anonymity refers to the fact that the attributes in dataset are modified in such a way that identity of individuals will not be revealed. This is essential in data mining especially the mining process is outsourced to a third party.



Zip	Gender	Age	Diagnosis
47918	Male	35	Cancer
47906	Male	33	HIV+
47918	Male	36	Flu
47916	Female	39	Obesity
47907	Male	33	Cancer
47906	Female	33	Flu

(a) Patient diagnosis records in a hospital

Zip	Gender	Age	Diagnosis
4791*	Person	[35-39]	Cancer
4790*	Person	[30-34]	HIV+
4791*	Person	[35-39]	Flu
4791*	Person	[35-39]	Obesity
4790*	Person	[30-34]	Cancer
4790*	Person	[30-34]	Flu

(b) The *k*-anonymity protected table of (a) when *k* = 3

Figure 2 – Shows normal data (a) and anonym zed data (b)

As can be seen in Figure 2, it is evident that the patient diagnosis data has some sensitive columns which are to be anonymized when the data is given for data mining. The anonymized data can be seen in figure 2 (b). There are three columns anonymized. Therefore this is called k-anonymity where k value is 3. More information on this can be found in [18]. Magkos et al. [19] employed election paradigm for privacy preserving data mining. They also discussed about its security requirements. Their approach proved to be effective as a building block which is used to obtain Random Forests Classification with accurate prediction performance. A good review of PPDM techniques were presented in [20], [21], [22] and [25].

Dasgupta et al. [23] introduced and discussed many metrics meant for measuring privacy and also explored to visualize the same. Chakravorty [24] explored PPDM techniques for smart homes. They also focused on data security and privacy in all data mining operations. Mogre et al. [26] applied PPDM techniques for high – dimensional data. Especially their focus was on anonymization techniques. Mogre and Patil [27] did similar kind of work and named the technique as SLICING which is an approach for handling high-dimensional data with privacy. Clifton et al. [29] explored tools which are used for PPDM. They discussed some methods for PPDM known as Secure Sum, Secure Set Union, and Secure Size for Set Intersection, and Scalar Product. The applications with which they employed the tools include association rule mining and EM clustering.

V. PPDM ON BIG DATA

Gosain and Chugh [28] explored PPDM methods for big data mining. As the traditional methods for PPDM are not sufficient for big data, new mining techniques are to be explored. In [28] three methods are discussed such as data anonymization, differential privacy and notice and consent. For data anonymization K-Anonymity [30], T-Closeness [31] and L-Diversity [30] are discussed with their limitations with respect to big data mining. Suppression and generalization can be used to achieve k-Anonymity. With respect to suppression, quasi identifiers are replaced using some constant values while the generalization replaces with general values. L-diversity brings about diversity in the sensitive attributes of given dataset. Differential privacy is another technique which is meant for reducing chances finding identity of individuals in the data. For this reason this is most useful technique which can be applied to big data as well. In this technique data is not modified as opposed to k-Anonymity. Moreover user has no direct access to database and it can act as a firewall which will preserve privacy. There are many advantages of using differential privacy. The original data need not be modified. Noise is added to results only. The distortion is done in such a way that the values are useful for analysis.

VI. PPDM RESULTS

As explored in [18] PPDM causes information distortion. For instance k-Anonymity can cause some information loss or distortion. Experimental results on the k value and the information distortion for Iris dataset and Wine dataset are presented in Figure 3.

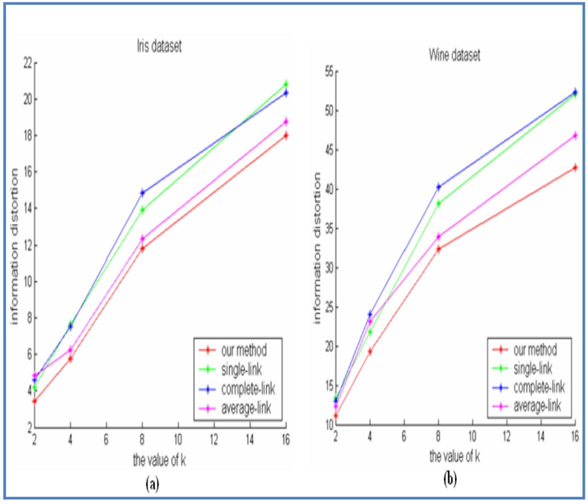


Figure 3 – Information distortion dynamics for Iris and Wine datasets (k-Anonymity results)

As can be seen in Figure 3, it is evident that the information distortion is there due to perturbation of data in order to preserve privacy [18]. The classification results of Random Forests as explored in [19] are presented in Figure 4.

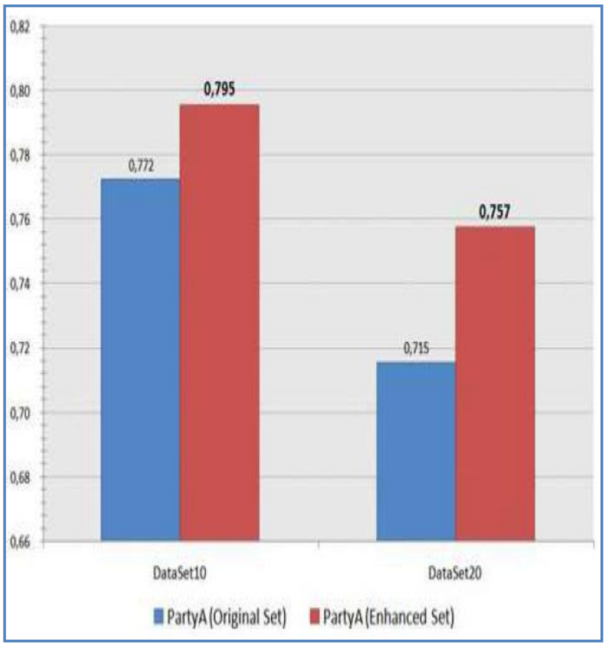


Figure 4 – Classification results using Random Forests

As can be seen in Figure 4, it is evident that the classification results reveal that the RF has better classification performance which is measured using F-measure which is the combined result of precision and recall metrics.

VII. CONCLUSIONS AND RECOMMENDATIONS

In this paper we study big data mining, distributed programming frameworks and the privacy preserving data mining for big data. Since big data refers to large volumes of data which is being accumulated in enterprises, processing such huge amount of data is called big data mining. Big data mining can get rid of biased conclusions as it can process data comprehensively and bring about business intelligence that can help enterprises to grow faster organically. Big data mining derives value from big data which is essential for expert decision making. Moreover big data mining needs a new programming paradigm known as MapReduce. Many frameworks came into existence that depends on MapReduce or such programming paradigm. They include Hadoop, Halopop, Sailfish, Dryad and AROM. When data is process in distributed environment like cloud computing, it is essential to ensure the privacy of the individuals that are associated with data being mined. Thus privacy preserving data mining techniques came into existing. This paper throws light into big data mining, distributed programming frameworks and the PPDM techniques. This research can be further extended to design and implement algorithms that can leverage the true parallel processing power of distributed programming frameworks in the real world besides preserving privacy.

VIII. REFERENCES

[1]. Dayton L. Jones, Kiri Wagstaff, David R. Thompson, Larry D'Addario, Robert Navarro, Chris Mattmann, Walid Majid, Joseph Lazio, Robert Preston, and Umaa Rebbapragada. (2012). Big Data Challenges for Large Radio Arrays. IEEE. p1-12.

[2]. Matthew Smith, Christian Szongott Distributed Computing & Security Group, Benjamin Henne and Gabriele von Voigt. (2013). Big Data Privacy Issues in Public Social Media. IEEE. p1-6.

[3]. Edmon Begoli, James Horey. (2012). Design Principles for Effective Knowledge Discovery from Big Data. IEEE. p215-218.

[4]. Stephen Kaisler, Frank Armour, J. Alberto Espinosa and William Money. (2013). Big Data: Issues and Challenges Moving Forward. IEEE. p995-1004.

[5]. Arjun Kumar , HoonJae Lee and Rajeev Pratap Singh. (n.d). Efficient and Secure Cloud Storage for Handling Big Data. IEEE. p162-166.

[6]. Xindong Wu, Xingquan Zhu, Gong-Qing Wu and Wei Ding. (2014). Data Mining with Big Data. IEEE. 26 (1), p97-107.

[7]. Hang Yang and Simon Fong. (2012). Countering the Concept-drift Problem in Big Data Using iOVFDT. IEEE. p126-132.

[8]. James M. Tien. (2013). Big Data: Unleashing Information. IEEE. p1.

[9]. Juhnyoung Lee. (2013). The Future of Enterprise Computing. IEEE. p1.

[10]. Avita Katal, Mohammad Wazid and R H Goudar. (2013). Big Data: Issues, Challenges, Tools and Good Practices. IEEE. p404-409.

[11]. Elisa Bertino. (2013). Big Data - Opportunities and Challenges. IEEE. p479-480.

[12]. Kshitij A Doshi Tao. Zhong, Zhongyan Lu, Xi Tang, Ting Lou and Gang Deng. (2013). Blending SQL and NewSQL Approaches Reference Architectures for Enterprise Big Data Challenges. IEEE. p163-170.

[13]. Chun-Chieh Chen, Kuan-Wei Lee, Chih-Chieh Chang, De-Nian Yang and Ming-Syan Chen. (2013). Efficient Large Graph Pattern Mining for Big Data in the Cloud. IEEE. p531-536.

[14]. Duncan Hodges and Sadie Creese. (2013). Breaking the Arc: Risk Control for Big Data. IEEE. p613-621.

[15]. Ningyuxin and Liyueling. (2013). How We could Realize Big Data Value. IEEE. p425-417.

[16]. Vassilios S. Verykios1, Elisa Bertino2, Igor Nai Fovino2 Loredana Parasiliti Provenza2, Yucel Saygin3, Yannis Theodoridis. (2004). State-of-the-art in Privacy Preserving Data Mining. SIGMOD. 33 (1), p50-57.

[17]. Xiaodan Wu, Yunfeng Wang, Chao-Hsien Chu, Fengli Liu, Ping Chen and Dianmin Yue. (2006). A Close Look at Privacy Preserving Data Mining Methods. PACIS. p167-173.

[18]. Chuang-Cheng Chiu and Chieh-Yuan Tsai. (2007). A k-Anonymity Clustering Method for Effective Data Privacy Preservation. Springer. p89-99.

[19]. Emmanouil Magkos, Manolis Maragoudakis, Vassilis Chrissikopoulos and Stefanos Gritzalis. (2009). Accurate and Large-Scale Privacy-Preserving Data Mining using the Election Paradigm. Ionian Universit. p1-27.

[20]. Amar Paul Singh and Ms. Dhanshri Parihar. (2013). A Review of Privacy Preserving Data Publishing Technique. IJERMT. 2 (6), p32-38.

[21]. Nivetha.P.R and Thamarai selvi.K. (2013). A Survey on Privacy Preserving Data Mining Techniques. IJCSMC. 2 (10), p166-170.

[22]. K. Srinivasa Rao & B. Srinivasa Rao. (2013). An Insight in to Privacy Preserving Data Mining Methods. CSEA. 1 (3), p100-104.

[23]. Aritra Dasgupta, Min Chen and Robert Kosara1. (2013). Measuring Privacy and Utility in Privacy-Preserving Visualization. COMPUTER GRAPHICS forum. p1-13.

[24]. Seema Kedar, Sneha Dhawale and Wankhade Vaibhav. (2013). Privacy Preserving Data Mining. IJARCCE. 2 (4), p1677-1680.

[25]. Antorweep Chakravorty, Tomasz Wlodarczyk, Chunming Rong. (2013). Privacy Preserving Data Analytics for Smart Homes. IEEE. p23-27.

[26]. Neha V. Mogre, Prof. Girish Agarwal and Prof. Pragati Patil. (2013). Privacy Preserving for High-dimensional Data using Anonymization Technique. IJARCSSE. 3 (6), p185-189.

[27]. NEHA V. MOGRE and SULBHA PATIL. (2013). SLICING: AN APPROACH FOR PRIVACY PRESERVATION IN HIGH-DIMENSIONAL DATA USING ANONYMIZATION TECHNIQUE. IEEE. p103-108.

[28]. Anjana Gosain and Nikita Chugh. (2014). Privacy Preservation in Big Data. IEEE. 100 (17), p44-47.

[29]. Chris Clifton, Murat Kantarcioglu, Xiaodong Lin and Michael Y. Zhu. (n.d). Tools for Privacy Preserving Distributed Data Mining. IEEE. 4 (2), p1-7.

[30]. J. Sedayao, “Enhancing cloud security using data anonymization”, White Paper, Intel Coporation.

[31]. N. Li, T. Li, S. Venkatasubramanian, "t-Closeness: Privacy Beyond k-Anonymity and l-Diversity**,** " IEEE 23rd International Conference on Data Engineering, 2007, pp. 106 - 115.

[32]. Intel. (2013). Planning Guide Getting Started With Big Data. Intel IT Center. 0 (0), p1-24

[33]. Yingyi Bu, Bill Howe, Magdalena Balazinska and Michael D. Ernst (2010).HaLoop: Efficient Iterative Data Processing on Large Clusters. USA: IEEE. p1-12.

[34]. SriramRao, Raghu Ramakrishnan and Adam Silberstein (2012). Sailfish: A Framework For Large Scale Data Processing. USA: Microsoft. p1-14.

[35]. Nam-Luc Tran and SabriSkhiri and Arthur Lesuisse and Esteban Zim´anyi (2012). AROM: Processing Big Data With Data Flow Graphs and Functional Programming. Belgium: Amazon. p1-8.