A Review of Big Data and Anonymization Algorithms

U. Selvi, Dr. S. Pushpa

1 Research Scholar- Department of Computer science and Engg, St.Peter’s University, Avadi, Chennai-
2 Professor- Department of Computer science and Engg, St.Peter’s University, Avadi, Chennai-
1slvunnikrishnan@gmail.com, 2pushpasangar96@gmail.com

Abstract

Over the past twenty years, information has raised in a very massive scale in various fields. In 2010, Apache Hadoop outlined BigData as “datasets that couldn't be captured, managed, and processed by general computers within a tolerable time”. This paper begins with the definition, background knowledge and challenges of BigData. Then it shows the relation of BigData with other related technologies, like Cloud computing, web of things, information centers, and Hadoop. Big Data system is decomposing into four phases; particularly generation, acquisition, storage, and analysis of massive information and this paper make a case for every section. Finally, this paper examines the security issues in Big Data and compares various anonymization algorithms. These discussions aim to produce a comprehensive summary of Big Data and its security.


I. DEFINITION AND FEATURES OF BIG DATA

Big Data can be defined as data sets that grow so large from several TB to ZB which cannot be managed by Traditional Database Management tools and it is difficult to capture, store, search, share, analyze and visualize data. Big Data include unstructured data and provides opportunities for discovering new values, understanding of unknown values, and gives as new challenges to effectively organize and manage such datasets.

Nowadays, big data related to the service of Internet companies grow rapidly. Big data also brings about many challenging problems which needs solution with the rapid growth of Cloud computing and Internet of Things (IOT). The increasingly growing data cause a problem of how to store and manage such huge heterogeneous and datasets with moderate requirements on hardware software infrastructure. This leads to challenge of collecting and integrating enormous amount of data. Mining the Datasets can help in decision making which reveal its intrinsic property.

Big data shall mean such datasets which could not be acquired, stored, and managed by classic database software. Big Data can be defined by 4V model [6], i.e., Volume, Velocity, Variety, and Value. In 4V model, Volume specifies generation and collection of large data; Velocity specifies timeliness of data collection and analysis; Variety specifies different types of data like structured, semi-structured and unstructured data; Value specifies hidden information from data.

1.1. Challenges of Big Data

The increase in Data size raises the technical issues in data acquisition, storage, management and analysis of Big Data. Relational RDBMs could not handle huge volume and heterogeneity of big data. Hence for permanent storage and management of large-scale disordered datasets, distributed file systems and NoSQL [3] databases are good choices other than traditional database systems.

These are some of the obstacles [4-5] in the development of Big data applications.
- Data representation: makes data more meaningful and used for analysis and user interpretation.
- Redundancy reduction and data compression: reduce the indirect cost of the entire system and the potential values of the data are not affected.
- Data life cycle management: decides which data shall be stored and which data shall be discarded. Since current storage system could not support such massive data.
- Data confidentiality: Big Data service providers or owners at present could not effectively maintain and analyze due to the huge size of data with limited capacity. They must rely on tools or third party to ensure safety.
- Energy management: system-level power consumption control and management mechanism is needed for big data. Since, electrical energy is consumed with the increase of data volume, processing, storage, and transmission. Hence
- Expendability and scalability: needs analytical system and algorithm of big data to process more complex and expanding datasets.
- Cooperation: Big data network architecture must be established to help scientists and engineers in various fields access different kinds of data and
fully utilize their expertise, so as to cooperate to complete the analytical objectives.

II. RELATED TECHNOLOGIES

There are several technologies closely related to Big data which include [6] Cloud Computing, IoT (Internet of Things), Hadoop, and Data centers. Let us see the relation of Big data with other related technologies.

2.1. Cloud computing and big data

Big data is about extracting value from the Information Assets, while Cloud focuses on On-Demand, Elastic, Scalable, pay-per use self service models. Big data need large on-demand compute power and distributed storage to crunch the 3V data problem and cloud provides this elastic on-demand compute required for the same. Cloud hides the complexity and challenges involved in building a scalable elastic self-service application. Big Data Hadoop in a similar way hides the complexity of the large scale distributed processing from the end user perspective which can be done by writing Map Reduce programs or Hive or Pig. Big data growth is mainly because of application demands and cloud computing from virtualized technologies. Cloud computing is a service and promote the growth of Big data.

2.2. IoT and big data

Huge amount of Information are generated from sensors devices and machines which include environmental data, geographical data, astronomical data, and logistic data. Mobile equipments, transportation facilities, public facilities, and home appliances could all be data acquisition equipments in IoT.

These data collected include heterogeneity, variety, unstructured feature, noise, and high redundancy. Data generated from IoT has three features that conform to the big data paradigm:
(i) Large amount of data are generated from abundant terminals;
(ii) Data generated by IoT is usually semi-structured or unstructured;
(iii) Data generated by IoT is useful only when it is analyzed with in a tolerable time.

It has been widely recognized that these two technologies are inter-dependent and should be jointly developed: on one hand, the widespread deployment of IoT drives the high growth of data both in quantity and category, thus providing the opportunity for the application and development of big data.

It also accelerates the research advances and business models.

2.3. Hadoop and big data

Hadoop is an open source framework for distributed storage and data-intensive processing. Presently, Hadoop is widely used in big data applications in the industry and in academic research. In addition, many companies provide Hadoop commercial execution and/or support, including Cloudera, IBM, MapR, EMC, and Oracle. CloudView, which is a framework for data organization and Cloud computing infrastructure, uses Hadoop to analyze machine generated data. Clusters based on Hadoop are used for complex offline analysis, e.g., case-driven data analysis.

2.4. Data center and Big Data

Data Centers is a platform for data storage and takes responsibility such as acquiring data, managing data, organizing data, and leveraging the data values and functions. Data centers support Big data however this are the following infrastructure that is desperately required:

i) Enterprise must develop data centers to improve the capacity of rapidly and effective processing of big data under limited price/ performance ratio.
ii) The growth of Big data applications accelerates a revolution so as to reduce the operational cost for the development of Data centers.
iii) Data centers should concerns with hardware facilities as well as with software facilities.

III. BIG-DATA SYSTEM ARCHITECTURE

In this section, we define the value chain for big data analytics which consist of four phases [6] i.e., generation, acquisition, storage, and processing. The details for each phase are explained as follows.

3.1. Data generation

In Big Data, data are generated from Enterprise, IoT, and Bio-medical application and from other fields. Enterprise data include production data, inventory data, sales data, and financial data, etc., Enterprise process one million to 75 million events per day for its target customer and advertisement. Given Internet data as an example, huge amount of data in terms of searching entries, Internet forum posts, chatting records, and micro blog messages, are generated IoT. Its network architecture may be divided into three layers: the sensing layer, the network layer, and the application layer. The sensing layer is responsible for data acquisition and mainly consists of sensor networks. The network layer is responsible for information transmission and processing, where close transmission may rely on sensor networks, and remote transmission shall depend on the Internet. Finally, the application layer support specific applications of IoT.
Big data also has Bio-medical data which include medical images and electronic health records, so that medical professionals may utilize this data to extract useful clinical information from masses of data to obtain a medical history and forecast treatment effects, thus improving patient care and reduce cost. Other field which includes computational Biology, astronomy Sloan Digital Sky Survey (SDSS), High energy physics, pervasive sensing and computing among nature, commercial, internet, government, and social environment are generating heterogeneous data.

3.2. Data acquisition

Data acquisition includes data collection, transmission, and data pre-processing. Data are collected from Log files, sensors, and Web Crawler. Log files are record file that automatically generate data in designated file formats. Ex: web server records number of clicks, visit and other property of web user.

Sensor data also includes sound waves, voices, vibration, chemical, current, pressure, temperature, etc., Web crawler acquires a URL in the order precedence through a URL queue and then downloads web pages, and identifies all URLs in the downloaded web pages, and extracts new URL to be put in the queue. These collected data are transmitted through wired or wireless networks. Data are transported to data storage infrastructure for processing and analysis. Data are mainly stored in data center and data transmissions are from data source to data centers or within data centers. Data centers consist of multiple integrated server racks interconnected with its internal connection networks. In Data preprocessing, data are integrated, cleaned and compressed to ensure consistency and to eliminate redundancy.

3.3. Data Storage

To ensure reliability and availability of data accessing, large scale datasets must be stored and effectively managed. Existing storage mechanisms for Big data is classified as file systems, database, and programming models. Google’s GFS, Colossus, Hadoop, Taobao TFS and FastDFS are some file system to support large-scale, distributed, data-intensive applications.

NoSQL[6] i.e., Non - traditional relational databases is becoming popular for Big data storage. Key-valued database (Ex: Amazon Dynamo DB), column-oriented databases (Ex: Big Table, Cassandra, HBaseand, HyperTable) and document oriented databases (Ex: MongoDB, SimpleDB, and CouchDB) are some of NoSQL database.

Programming model[6] for Big Data include MapReduce, Dryad and Pregel.

3.3.1. MapReduce

MapReduce[11] is a scalable and fault-tolerant data processing framework that enables to process huge volume of data in parallel. Map Reduce has two primitive functions namely Map and Reduce. The Map function generates intermediate key-value pair based on input key-value pair. This intermediate values are merged having the same key and transmitted to Reduce function, which further compresses the values to smaller set. It has been widely adopted and received extensive attention from both the academia and the industry because of its promising capability. MapReduce framework becomes more scalable and cost-effective because infrastructure resources can be provisioned on demand. Simplicity, scalability, and fault tolerance are three main salient features of MapReduce framework. Therefore, it is convenient and beneficial for companies and organizations to process big data and obtain core competiveness.

3.3.2. Dryad

Dryad[11] is a directed acyclic graph, in which vertices represent program and edges represent data channels. It is a framework to form data channels between programs in graph and handle them as data sets in parallel. Dryad developers design data processing in graph.

3.3.3. Pregel

Pregel[11] provide analysis of networks graphs and Social networking services. It computes over large graphs much faster than alternatives, and the application programming interface is easy to use. Pregel programming model consist of a sequence of iterations, called supersteps. During a supersteps the framework invokes a user defined function for each vertex, conceptually in parallel. It can read messages sent to vertices, send messages to other vertices and modify the state of vertices.

3.4. Big Data Analysis

To bring values and extract useful information from large data set, Big Data uses the following methods Bloom Filter, Hashing, Trie and Parallel computing. Bloom filter uses Hash values of data to provide compression and storage of data while Hashing convert the data to fixed length size. Trie utilize common prefixes of character strings to reduce compression on character string to improve query efficiency. In parallel computing, computing resources are simultaneously utilized to complete a computation task. This analysis can be done in Memory-level, Business Intelligence and Massive-level. R (30.7%), Excel (29.8%), Rapid-I Rapid miner (26.7), KNMINE (21.8%) and Weka/Pentaho (14.8%) are the various tools available for Big Data analysis.
IV. BIG DATA APPLICATION

Big data application includes various fields like
- Enterprise
- IoT
- Online social networks
- Health Care and medicals
- Collective Intelligence
- Smart grid

V. BIG DATA SECURITY

Big data privacy includes two aspects:

(i) Protection of personal privacy during data acquisition: personal interests, habits, and body properties of users may be more easily acquired, and users may not be aware.

(ii) Personal privacy data may also be leaked during storage, transmission, and usage, even if acquired with the permission of users. To ensure privacy, source data must be processed for anonymization.

5.1. Anonymization algorithm

Data anonymization refers to concealing identity and sensitive data for owners of knowledge records. Then, the privacy is preserved whereas aggregate data is exposed for information users for analysis and mining.

Many Anonymization algorithms are listed in the literature [7-10] as follows:

5.1.1. K-anonymity

In k-anonymity, the information of a personal during revealed information cannot be unambiguously known from a minimum of k – 1 people there in revealed data. QID should appear in at least k records. For example, if a released table which has Birth data and gender as QID, then k-men in the table should have the same Birth date and gender to achieve k-anonymity. In a k-anonymous table, each record is indistinguishable from at least k-1 other records with respect to QID. K-anonymity can be achieved by generalization and suppression. Generalization can be achieved by Replace specific quasi-identifiers with less specific values until “k” identical values are identified and Partition ordered-value domains into intervals, whereas suppression is needed when generalization causes too much information loss.

Depending on various background knowledge assumptions with the attacker, different authors have presented different variants of k-anonymity as follows:

i) k-candidate anonymity: to ensure anonymity the individual ought to have a minimum level of uncertainty concerning the re-identification of any node within the graph (Hay et al., 2007). This will be achieved by k-candidate anonymity that may be a generalization of k-anonymity planned by Sweeney (2002). K-Candidate anonymity says that any person cannot be re-identified from k alternative people there in graph.

ii) k-degree anonymity: In k-degree anonymity, during which each node has a minimum of k – 1 alternative node within the graph having same degree. This prevents re-identification of people by adversaries having previous information of degree of bound nodes. The authors have outlined a graph anonymisation drawback by which, given a graph G, asks for k-degree anonymous graph that stems from G, with minimum number of graph modifications [12].

iii) k-neighbourhood anonymity: If associate degree individual has some information concerning the neighbours of a target victim and therefore the relationship among the neighbours, the victim could also be re-identified during a social network albeit the victim’s identity is preserved.

5.1.2. Randomisation

Ying and Wu (2008) noted that graph perturbation provides protection from structural attack. However it introduces structural changes ensuing data loss. Ying and Wu (2008) planned Randomisation techniques for privacy protection in a social network graph:

i) Random add/del technique: In this technique, random false edged are added to the network graph followed by identical range of true edge deletion specified range of edges stay unchanged. Ying et al. (2009) compared this approach with k-anonymity and located that random add/del technique provides protection from each identity and link revealing whereas k-anonymity through preserves a lot of structural properties will defend solely against identity revealing.

ii) Random switch edges: In this technique, edges are swapped and specified nodes degree stay unchanged. the k-anonymity and randomization

13128
mentioned higher than cater just for identity and link disclosures during a social network.

5.1.3.  l-diversity

Sensitive attributes must be diverse within each QID equivalence class. So as to realize l-diversity in social network, Zhou and Pei (2010) introduced l-diverse partition wherever vertices have to be compelled to be partitioned off into equivalence teams, specified in each equivalence cluster of vertices, at {the most} 1/l of the vertices are related to the most frequent sensitive labels. The author has outlined l-diverse partition in social networks. Given a social network G = (V, E) with n vertices and every vertex is related to a non-sensitive label and a sensitive label, associate degree l-diverse partition divides the vertices V into m equivalent teams of vertices. specified \( \frac{\text{freq}(c)}{|E_G|} \leq 1/l \) wherever freq(c) is that the range of vertices that carry the foremost frequent sensitive label c in cluster EG, and | EG | is that the range of vertices within the corresponding equivalence cluster. L-diversity is not sufficient to provide privacy protection, when the overall distribution of the sensitive attribute is skewed. L-diversity does not consider semantics of sensitive value.

5.1.4.  t-closeness

T-closeness distribute sensitive attributes within each quasi-identifier group should be “close” to their distribution in the entire original database. t-closeness uses the Earth Mover Distance (EMD) operate to live the closeness between two distributions of sensitive values and needs the closeness to be at intervals. It primarily specialize in categorical sensitive attributes.

5.1.5.  Multi-dimensional Mondrian Algorithm

The Mondrian is a greedy partitioning algorithmic program wherever in, anonymization is generated in two sections. The Mondrian [12] is a greedy partitioning algorithmic program wherever in, anonymization is generated through two phases. The first phase defines multi-dimensional regions within the domain area recursively till all regions have a minimum of k information things. The second section applies recording functions for every region mistreatment some outline statistics. Two ways for partitioning and recording are delineated as: strict, relaxed partitioning and global, native recording severally. A strict partitioning defines a collection of non-overlapping and relaxed partitioning a potentially overlapping multi-dimensional region. Global and local cryptography seeks to realize k-anonymity by severally mapping, the domains of the quasi-identifier attributes to generalized or altered values and individual instances of knowledge items to generalized values.

5.1.6.  Differential privacy

The notion of differential privacy was developed as a principled way of defining privacy, so that the risk to one’s privacy should not substantially increase as a result of participating in a database. This shifts the view on privacy from comparing the prior and posterior beliefs about individuals before and after publishing a database to evaluating the risk incurred by joining a database. It also imposes a guarantee on the data release mechanism rather than on the data itself. Here, the goal is to provide statistical information about the data while preserving the privacy of users in the data. This privacy definition gives guarantees that are independent of the background information and the computational power of the adversary.

For example, if the social network data set is released using a differentially private mechanism and A is a member in social network, this would guarantee that A’s participation in the social network does not pose a threat to A’s privacy because the statistics would not look very different without A’s participation. It does not guarantee that one cannot learn sensitive information about A using background information but such guarantee is impossible to achieve for any kind of dataset.

5.1.7.  k-automorphism anonymity

An anonymized graph is k-automorphic if every node in the graph has the same subgraph signature (of arbitrary size) as at least k − 1 other graph nodes, and the likelihood of every candidate for that node is less than or equal to 1/k.

5.1.8.  Comparison of Algorithm

<table>
<thead>
<tr>
<th>Anonymization Algorithm used</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>K-anonymity</td>
<td>Protect against Record Linkage and attribute linkage attacks [2]</td>
<td>Sensitive attributes and back ground knowledge attack</td>
</tr>
<tr>
<td>l-diversity</td>
<td>Protect against sub graph and attribute linkage attacks</td>
<td>Large data utility loss and skewness attack</td>
</tr>
<tr>
<td>Randomization</td>
<td>Protect against Identity and link disclosure</td>
<td>Degrade data utility</td>
</tr>
</tbody>
</table>
VI. CONCLUSION

In this paper, background knowledge of Big data and its related technologies has been reviewed. Then, this paper focuses on four phases of Big Data i.e., generation, acquisition, storage, analysis of big Data and its technical challenges. Finally, it addresses the security issues of Big Data and compares various Anonymization algorithms.

A nature of data presents requirements of scalability and speed that cannot be satisfied through current research. The privacy preservation for data analysis, sharing and mining is a challenging research issue due to increasing volumes of datasets and further research is needed in this direction.

REFERENCES

1. Mohamed R.Fouad, Khaled Elbassioni, and Elisa Bertino, A Super Modularity Based Differential Privacy Preserving Algorithm for Data Anonymization IEEE Transactions on Knowledge and Data Engineering Vol.26, No 7 July 2014