Big Data Anonymization with Spark

Yavuz Canbay
Department of Computer Engineering
Faculty of Engineering, Gazi University
Ankara, Turkey

Seref Sağıroğlu
Department of Computer Engineering
Faculty of Engineering, Gazi University
Ankara, Turkey

***Abstract*—Privacy is an important issue for big data including sensitive attributes. In the case of directly sharing or publishing these data, privacy breach occurs. In order to overcome this problem, previous studies were focused on developing big data anonymization techniques on Hadoop environment. When compared to Hadoop, Spark facilitates to develop faster applications with the help of keeping data in memory instead of hard disk. Despite a number of projects were developed on Hadoop, now this trend is shifting to Spark. In addition, the problem of anonymizing big data streams for real-time applications can be solved with Spark technology. Hence to sum up, Spark is the main technology facilitates developing both faster anonymization applications and big data stream anonymization solutions. In this study, anonymization techniques, big data technologies and privacy preserving big data publishing was reviewed and a big data anonymization model based on Spark was proposed for the first time. It is expected that the proposed model might help to researchers to solve big data privacy issues and also provide solutions for new generation privacy violations problems.**

Keywords—big data, anonymization, privacy preserving, hadoop, spark, model, review

I. INTRODUCTION

Big data is a new popular data paradigm. Basically, increase in volume, variety and velocity of data lead people to meet the new requirements. Because, traditional systems, solutions and technologies are not capable of processing big data. With the help of a distributed and large scale processing, big data architecture presents powerful solutions to the big problems. Hence, big data problems can be solved via existing and developing big data technologies. Companies, organizations, institutions and also governments spend great efforts on big data in order to shape their politics and future plans. Big data is seen as a promising concept for solutions between different domains by enabling faster, deeper, more accurate and significant inferences to help making better decisions, shaping future plans and developing innovative politics [1, 2].

Privacy of big data gains importance with the need of processing big data including sensitive attributes. It is also a confidentiality statement for big data including personal information and defines using big data in a privacy border for individuals [1]. It is a major issue requiring strong preservation and attracting great attention from data holder [3]. Some kind of big data such as medical, financial, social, etc. often contains personal sensitive data such as identity number, age, name, surname, sex, etc. that can be a malicious tool to disclose an individual [4]. For example, a health record in the database can reveals record owner's disease or a census record can uncover owner's salary. Hence, when dealing with big data containing sensitive information, privacy must be the first concern to be handled. It should be known that, deleting some attributes such as name, surname, identity number or any others from data, is not an adequate measure to preserve privacy. The researches have shown that linkage of some attributes such as zip code, date of birth and sex, which locates in different databases, identifies %87 of individuals in US uniquely [5]. This case proves that data privacy needs to be handled in a deeper and wider context.

In order to obtain big values from big data, sometimes publishing it to the public or sharing it with stakeholders play an important role. For example, a data holder may want their data to be analyzed by third parties to obtain hidden patterns, an institution may open weather forecast data to public and an organization may want to predict future attacks by sharing its network flow data with security companies. Generally speaking, obtaining maximum benefits from big data depends on sharing or publishing it. While publishing of big data brings some significant benefits, it also reveals some privacy issues [6-11]. Hence, publishing of big data requires the concept of privacy preserving. Privacy preserving big data publishing is still an important problem for researchers. Because, developing new privacy techniques compatible with big data architecture and data variety are some concerns that researches must focus on.

Hadoop and Spark are two fundamental big data technologies. Hadoop provides processing data on disk while Spark process data on memory. Spark runs 100 times faster than Hadoop. This difference plays an important role for some projects requiring short response time. In addition, Spark is the major technology used in streaming big data analytics. Both technologies have a distributed data processing framework called as MapReduce. It distributes a costly process among multiple nodes and thus it provides a performance enhancement with parallel computation [7, 12].

The main contribution of this paper is that it proposes a model that suggests developing anonymization techniques based on Spark for faster anonymization processes. Previous works evaluate Hadoop based big data privacy preserving applications. But this study evaluates big data privacy with a Spark based anonymization model to take Spark's advantages which were mentioned above. This paper was organized as

978-1-5386-0930-9/17/$31.00 ©2017 IEEE

follows. Section II presents preliminary knowledge about the main topics. In Section III, a literature review was summarized. The proposed model was explained in Section IV. Section V gives conclusion remarks and future works.

II. PRELIMINARY

A. Anonymization

Anonymization is the process of hiding true identity of the data owner [1]. It generalizes or falsifies the data to be used in data mining or analyzing processes and provides acceptable solutions via minimizing disclosure risk [6, 13]. Although anonymization has been seen as a one-time process, the nature of the data always changing [12]. When this situation is evaluated with regard to streaming data, it can be seen that anonymization process is always necessary for continuous data and new data is being collected continuously. In addition, some user-interactive applications require short-time responses. Hence speed can be a discriminator factor for instant responses.

In a sensitive data table, each record can be including four type of attribute explained below and exemplified in Fig. 1.

- Identifier (ID): attributes that uniquely identify a person.
- Quasi-identifier (QID): attribute set with minimum element identifies a person with linking other datasets by combining the attributes.
- Sensitive (S): private attributes have the potential for violating record owner's privacy
- Non-sensitive (NS): non-private attributes violate record privacy of record owner.

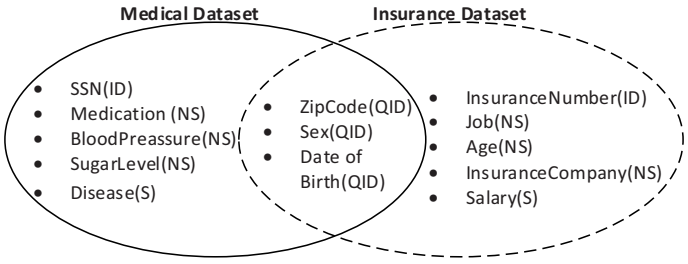

Fig. 1 ID, QID, S and NS attribute examples with linking records

In general, there are some basic anonymization techniques that facilitate to develop anonymization algorithms. Well known techniques are listed and briefly explained below [14].

- Generalization: replacing some values of QID's with its parent values located in the taxonomy tree. In another words, it means that replacing a specific value with a less specific or an upper level value. For instance, engineer or lawyer can be generalized to professional.
- Suppression: replacing some values with some special characters such as "*". For example, "1988" can be suppressed to "19**".
- Anatomization: splitting sensitive table into two tables which the first one includes QID attributes and the second one includes S attributes. Both tables include one common attribute which generally named as GroupID. The further processes are performed based on GroupID.
- Permutation: partitioning data into groups and shuffling their sensitive values within each group.
- Perturbation: original values of data are replaced with synthetic values with protecting of statistical information or distribution. The perturbed data includes synthetic data instead of real data.

As indicated in [12], the relation between privacy and utility is a key feature. While the original data presents a high utility, it reduces with anonymization. Hence it must be considered widely that while preserving privacy of data, the utility must be kept in an acceptable rate. The balance between privacy and utility must be kept optimally.

k-anonymity, l-diversity and *t-closeness* are the basic anonymization models. *k-anonymity* [12] is a privacy model that requires each record to be indistinguishable from at least k-1 records within published data even an attacker knows the values of QID's of the victim. It provides a solution for record linkage attack. *l-diversity* [12] is another privacy model that was developed to fill the deficiency of k-anonymity. Although *k-anonymity* provide privacy, it is fragile to preserve the privacy of sensitive attribute if the attacker knows the values of QID's. The main issue in here is the lack of the diversity of sensitive information in each equivalence classes. *l-diversity* also solves this problem by proposing each equivalence class must have at least *l well-presented* sensitive information. In addition, it solves both record linkage and attribute linkage problems. *t-closeness* [5] was developed to provide the distribution of the values of sensitive attribute in any equivalence class to be close to the distribution of the values of sensitive attribute in the entire table. Hence the problem of attribute linkage and probability attacks are solved.

B. Big Data

Big data is a term used for large, massive, complex data that cannot be handled by traditional systems in an acceptable time and exceeds the limits of existing computational capabilities of ordinal systems [15-17]. It also appeared in the literature in the age of when the digital world faced with some difficulties or barriers such as managing, storing, analyzing, retrieving, and visualizing of fast growing data [13]. Big data provides big opportunities to obtain big values from the data.

In the literature, big data is generally explained with 7V's. Volume describes huge amount of data (terabyte, petabyte, exabyte, vs.), Velocity is speed of input and output data (batch, streaming, real-time), Variety is used for diversity of

data type (structured, unstructured and semi-structured), Veracity presents truthfulness of the data, Value indicates useful outcomes, Variability refers to changes in data format, structure and semantics, and Visualization describes presentation of big data [3, 4, 17-19]. 7 V's of big data is presented in Fig. 2.

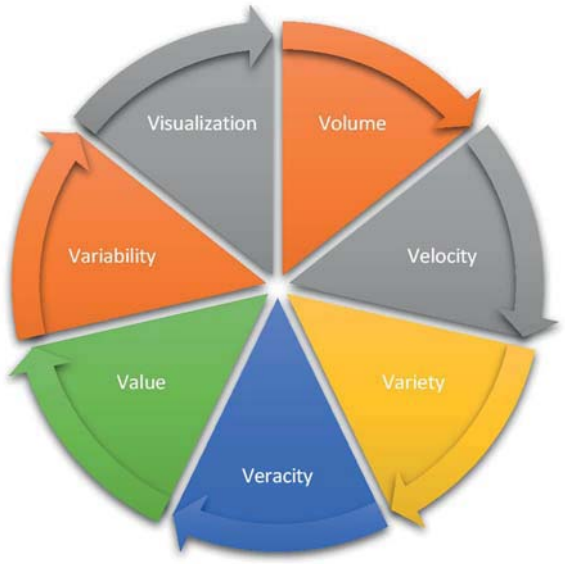

Fig. 2 7V's of big data

C. Spark

Spark is an open source project developed by Apache in order to handle big data. It provides a distributed computing platform which enables fast, efficient, fault-tolerant and scalable processing large, complex and massive data. This framework has some basic components such as Spark SQL, Spark Streaming, MLib and GraphX that makes it more powerful. Spark SQL is used to query data with SQL, Spark Streaming facilitates stream processing, MLib is a library includes machine learning algorithms and GraphX is used for graph analytics [20].

Resilient Distributed Datasets (RDDs) are distributed memory abstraction facilitates programmers to perform in-memory computations on large clusters to take the advantage of the speed [21]. As it was expressed in [20], in some projects, speed, ease of use and performance can be a reason for preference of Spark technology. For example, Spark is up to 100 times faster than Hadoop in memory and 10 times on disk. It provides developing algorithms based on MapReduce which consists of two main steps, Map and Reduce. They are designed by the application programmer based on what is desired to be done in each step. But the output of these two steps must be appropriate with the following definitions. Each Map task produces (key, value) pairs by processing its own chunk and each reducer combines all the values associated with a specified key. In Fig. 3, Apache Spark ecosystem is presented.

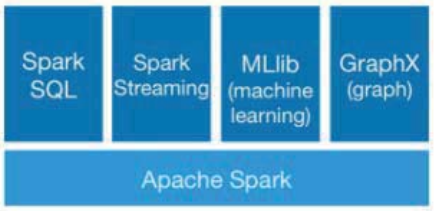

Fig. 3 Spark ecosystem [22]

III. Literature Review

In [1], it was emphasized that determining quasi-identifiers by manual is prone to human error. Hence a two-phase entropy based approach was proposed in this study. In the first phase, appropriate quasi-identifier set was found and in the second phase dataset with determined appropriate quasi-identifier set was anonymized. Classification accuracy and information loss were selected as performance criteria. Appropriate quasi-identifier set was selected based on entropy and cell generalization with minimum information loss was performed. Naïve Bayes was used to classify the data and k-means was utilized for clustering. The effect of quasi-identifiers on whole data set was analyzed. The attribute with low entropy means high risk to identify the value of sensitive attribute. Hence, the attributes having high entropy were considered as important to anonymize.

A privacy preserving big data publishing approach was proposed in [12]. Scalability problem of data anonymization was solved with the help of distributed programming framework, Hadoop. k-anonymity and l-diversity techniques were developed based on Hadoop, PokerHand dataset and a synthetic dataset were used to evaluate the performance of the developed algorithms. Information loss, k and l parameters, run-time and data transfer were chosen as performance metrics. The developed techniques were compared with MRTDS and Baseline.

In [6], a scalable two phase top-down specialization approach using MapReduce was proposed. The approach was developed on Hadoop and the performance was evaluated on Adult dataset. In the first phase, dataset was partitioned into smaller sets. Then intermediate anonymization levels were obtained with the anonymization of partitioned data. In the second phase, all the intermediate levels were merged into one and this merged part was anonymized to achieve k-anonymity. Run-time, data size, k-parameter, information loss and the number of partitions were used as performance criteria. The proposed method was compared to CentTDS approach and the performance of proposed method outperformed.

A scalable local recoding big data anonymization approach using locality sensitive hashing and MapReduce was presented in [7]. A semantic distance metric was developed to measure similarity between two records. Data was recursively split into smaller partitions by using local sensitive hashing. Local sensitive hashing based k-member clustering algorithm was employed on these partitions in parallel. The performance of the proposed method was evaluated on Adult dataset. Performance tests were performed according to run-time, number of record, information loss and number of node.

Greedy k-member clustering algorithm was selected as a performance comparison technique

An application of big data anonymization on Hadoop was performed in [15]. A specific dataset was used to evaluate Top Down Specialization method. k-anonymity method was employed to provide anonymity. The anonymization process includes two phases. In the first phase, big data set was split into small partitions and then anonymization of each part. These anonymized partitions were then combined and merged into one huge data in the second phase. Information loss was used as a performance metric.

Local recoding anonymization approach for Big Data using MapReduce was developed in [8]. They presented a model that considers the problem of big data local recoding against proximity privacy breaches and proposed a scalable two phase clustering approach. A two phase approach including t-ancestor clustering algorithm and proximity-aware agglomerative clustering algorithms was developed. Adult dataset was used and the performance metrics were chosen as run-time, number of record and number of node.

Multidimensional big data anonymization approach using MapReduce was proposed in [3]. A high scalable median finding algorithm was developed. Firstly, the problem of finding median of numerical attribute of the data was examined and then the granularity of recursion of multidimensional plan in MapReduce was investigated. Mondrian which is a multidimensional anonymization algorithm was extended in this study based on MapReduce framework. Adult dataset was used to evaluate the performance of the developed algorithm. Run-time and data size were selected as performance criteria.

In [13], a multidimensional sensitivity based big data anonymization approach was proposed The approach extended k-anonymity concepts and named as multidimensional sensitivity based anonymization method. It was integrated with role based access control mechanism. The proposed approach was developed based on MapReduce programming paradigm. Adult dataset was used to evaluate the approach and the number of equivalence class and number of records was selected as performance criteria.

A two phase top down specialization based big data anonymization approach was proposed in [23]. The proposed method was developed by using MapReduce and it was evaluated using Adult dataset. The number of partition and run-time were used as performance criteria.

Top down specialization and bottom up generalization for big data anonymization was hybridized in [10]. MapReduce programming paradigm and Adult dataset were employed. It was emphasized that both traditional top down specialization and bottom up generalization methods suffers from parallelization, efficiency and scalability. Hence to overcome these problem these two approaches were developed using MapReduce. Sub-tree anonymization was performed via the hybrid method. Run-time and k parameter of k-anonymity were used as performance metrics. The hybrid method outperformed two of traditional methods.

IV. The Proposed Model

In this study, a general Spark based big data anonymization model was proposed. The model is an outline/guideline for a detailed anonymization process to be developed. The model uses Spark for a base architecture to take the advantages of speed. This model consists of three main and some sub steps. These steps are given and explained below in details.

Step 1. Read data from HDFS: reading the original data blocks from HDFS

Step 2. Spark based anonymization

- Basic spark processes: taking the input from Step 1, creating RDD from the input and then transforms it into new RDDs.
- Anonymization with Spark: responsible for anonymization process including map, reduce and some anonymization algorithm based decisions. In Fig 4, the enclosed area with dashed line represents this step. It is also a symbolic and general representation of anonymization. Due to a great number of anonymization techniques exist, there will be a specific logic based on anonymization algorithm for this field.

Step 3. Write data to HDFS: writes the desired anonymized output file to HDFS.

The flow in the proposed model start with reading data blocks from HDFS. Then RDD creation process was performed by Spark Context. With the help of a transformation process, new RDDs are created. Later than, MapReduce jobs designed for a specific anonymization technique, for instance k-anonymity, l-diversity, t-closeness, etc., were performed to anonymize data. The current state of anonymization and other intermediate processes are evaluated in a middle layer. Then if a predefined criterion is met, the model produces the desired anonymized data, otherwise it returns to new MapReduce jobs. The proposed model was presented in Fig. 4.

The proposed model presents a guideline for further detailed Spark based anonymization techniques. It draws a general overview how and where to start big data anonymization using Spark and how the sub processes of the anonymization should be designed. To the best of our knowledge, there is no model in the literature considering Spark based anonymization. If it is compared to the existing approaches developed on Hadoop and presented in Section III, the proposed model might offer faster anonymization applications due to the nature of Spark. In addition, while a privacy preserving big data mining approach with k-means and differential privacy based on Spark was proposed in [24], this paper focuses on a model based on privacy preserving big data publishing techniques, such as k-anonymity, l-diversity, t-closeness, etc. on Spark.

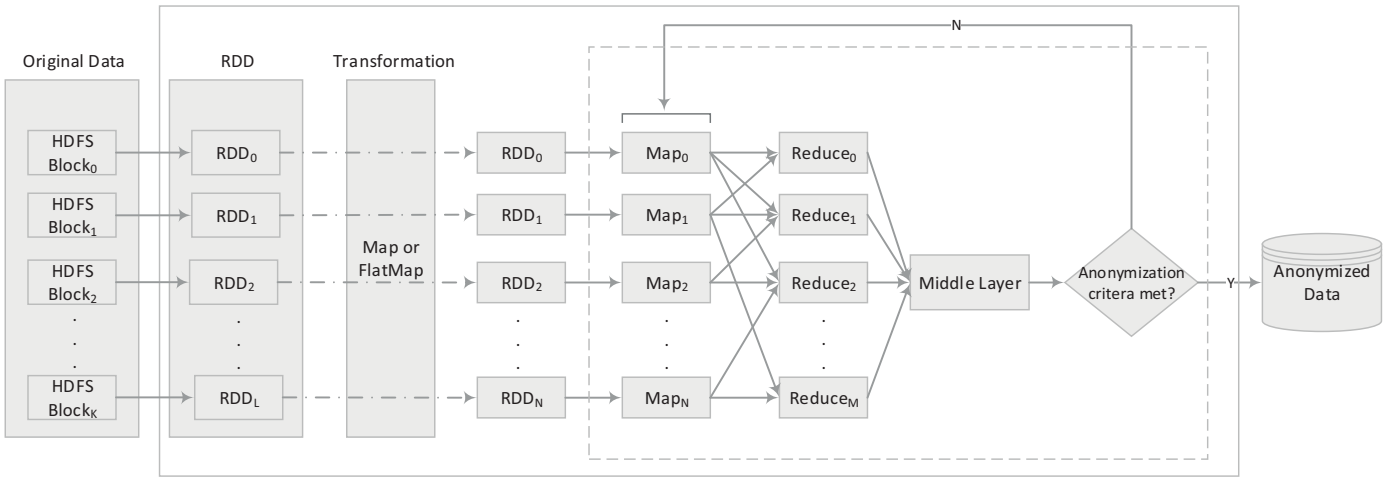

Fig.4. The Flowchart of the Proposed Model

V. Conclusion And Future Works

In this paper, privacy preserving big data publishing was reviewed and a Spark based model was proposed. A model including reliable, scalable and faster solution for big data anonymization was achieved with the help of Spark. This model's MapReduce jobs can be modified according to the preferred anonymization technique to be developed. Hence the proposed model provides a guideline for a faster anonymization of big data using Spark. Beside speed, Spark provides to process streaming big data. As a results, anonymization of streaming big data problem can be solved.

Anonymization of big data brings some challenges such as scalability (handling huge number of data to be anonymized), performance (balancing utility and privacy), timeliness (anonymizing big data in an acceptable time), dimensionality (handling large number of attributes) and so on. The proposed model also minimizes these challenges by using Spark.

Anonymization is not just deleting some attributes, replacing values with other and etc. It is finding an optimal tradeoff between privacy and utility. Hence this situation also makes the problem more challenging.

Anonymization is a big problem when the data becomes big. Because processing and anonymizing high-dimensional and large scale data, streaming data, huge amount of data, various type of data is a challenging problem.

Finally, it can be concluded that anonymization might be also achieved on Spark and this helps to develop faster anonymization applications in comparison with Hadoop.

In the future, we are planning to build up this model and test it. In addition, we will improve some anonymization algorithms considering both nominal and numeric attributes. Also we expect that the developed model will be used not only in our projects but also in other projects as well.

REFERENCES

[1] A. Ranjan and P. Ranjan, "Two-phase entropy based approach to big data anonymization," in *Computing, Communication and Automation (ICCCA), 2016 International Conference on*, 2016, pp. 76-81.

[2] D. S. Terzi, R. Terzi, and S. Sagiroglu, "A survey on security and privacy issues in big data," in *Internet Technology and Secured Transactions (ICITST), 2015 10th International Conference for*, 2015, pp. 202-207.

[3] X. Zhang, C. Yang, S. Nepal, C. Liu, W. Dou, and J. Chen, "A MapReduce based approach of scalable multidimensional anonymization for big data privacy preservation on cloud," in *Cloud and Green Computing (CGC), 2013 Third International Conference on*, 2013, pp. 105-112.

[4] W. Li and H. Li, "LRDM: Local Record-Driving Mechanism for Big Data Privacy Preservation in Social Networks," in *Data Science in Cyberspace (DSC), IEEE International Conference on*, 2016, pp. 556-560.

[5] B. C. Fung, K. Wang, A. W.-C. Fu, and S. Y. Philip, *Introduction to privacy-preserving data publishing: Concepts and techniques*: CRC Press, 2010.

[6] X. Zhang, L. T. Yang, C. Liu, and J. Chen, "A scalable two-phase top-down specialization approach for data anonymization using mapreduce on cloud," *IEEE Transactions on Parallel and Distributed Systems,* vol. 25, pp. 363-373, 2014.

[7] X. Zhang, C. Leckie, W. Dou, J. Chen, R. Kotagiri, and Z. Salcic, "Scalable Local-Recoding Anonymization using Locality Sensitive Hashing for Big Data Privacy Preservation," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 2016, pp. 1793-1802.

[8] X. Zhang, W. Dou, J. Pei, S. Nepal, C. Yang, C. Liu, *et al.*, "Proximity-aware local-recoding anonymization with mapreduce for scalable big data privacy preservation in cloud," *IEEE transactions on computers,* vol. 64, pp. 2293-2307, 2015.

[9] Y. Qu, J. Xu, and S. Yu, "Privacy preserving in big data sets through multiple shuffle," in *Proceedings of the Australasian Computer Science Week Multiconference*, 2017, p. 72.

[10] X. Zhang, C. Liu, S. Nepal, C. Yang, W. Dou, and J. Chen, "A hybrid approach for scalable sub-tree anonymization over big data using MapReduce on cloud," *Journal of Computer and System Sciences,* vol. 80, pp. 1008-1020, 2014.

[11] K. Xu, H. Yue, L. Guo, Y. Guo, and Y. Fang, "Privacy-preserving machine learning algorithms for big data systems," in *Distributed Computing Systems (ICDCS), 2015 IEEE 35th International Conference on*, 2015, pp. 318-327.

[12] H. Zakerzadeh, C. C. Aggarwal, and K. Barker, "Privacy-preserving big data publishing," in *Proceedings of the 27th International Conference on Scientific and Statistical Database Management*, 2015, p. 26.

[13] M. Al-Zobbi, S. Shahrestani, and C. Ruan, "Sensitivity-based anonymization of big data," in *Local Computer Networks Workshops (LCN Workshops), 2016 IEEE 41st Conference on*, 2016, pp. 58-64.

[14] A. Mehmood, I. Natgunanathan, Y. Xiang, G. Hua, and S. Guo, "Protection of big data privacy," *IEEE access,* vol. 4, pp. 1821-1834, 2016.

[15] N. P. KS and T. Pratheek, "Providing anonymity using top down specialization on Big Data using hadoop framework," in *India Conference (INDICON), 2015 Annual IEEE*, 2015, pp. 1-6.

[16] H. K. Patil and R. Seshadri, "Big data security and privacy issues in healthcare," in *Big Data (BigData Congress), 2014 IEEE International Congress on*, 2014, pp. 762-765.

[17] N. Victor, D. Lopez, and J. H. Abawajy, "Privacy models for big data: a survey," *International Journal of Big Data Intelligence,* vol. 3, pp. 61-75, 2016.

[18] I. Olaronke and O. Oluwaseun, "Big data in healthcare: Prospects, challenges and resolutions," in *Future Technologies Conference (FTC)*, 2016, pp. 1152-1157.

[19] M. Tanwar, R. Duggal, and S. K. Khatri, "Unravelling unstructured data: A wealth of information in big data," in *Reliability, Infocom Technologies and Optimization (ICRITO)(Trends and Future Directions), 2015 4th International Conference on*, 2015, pp. 1-6.

[20] L. Hbibi and H. Barka, "Big Data: Framework and issues," in *Electrical and Information Technologies (ICEIT), 2016 International Conference on*, 2016, pp. 485-490.

[21] M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. McCauley, *et al.*, "Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing," in *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, 2012, pp. 2-2.

[22] (06.06.2017). *Apache Spark*. Available: http://spark.apache.org/

[23] S. Kavitha, S. Yamini, and R. Vadhana, "An evaluation on big data generalization using k-Anonymity algorithm on cloud," in *Intelligent Systems and Control (ISCO), 2015 IEEE 9th International Conference on*, 2015, pp. 1-5.

[24] Z.Q. Gao and L.J. Zhang, "DPHKMS: An Efficient Hybrid Clustering Preserving Differential Privacy in Spark," in *International Conference on Emerging Internetworking, Data & Web Technologies*, 2017, pp. 367-377.