An Effective Security and Privacy in Big Data analytics Using Machine Learning

Shrihari M R[1], Dr. Manjunath.T.N[2]
[1]*Department of CSE, SJCIT Chickballapur*
[2]*Department of ISE, BMSIT&M Bangalore*

Abstract- **The Conservation of privacy largely relies on technological limitations on the ability to extract, analyze, and correlate potentially sensitive data sets. However, advances in Big Data analytics provide tools to extract and utilize this data, making violations of privacy easier. As a result, along with implementing Big Data tools, it is necessary to create safeguards to prevent abuse In addition to privacy; data used for analytics may include regulated information or intellectual property. Moreover, current efforts aimed at improving and extending Map Reduce to address identified challenges are presented. Accordingly, by identifying issues and challenges Map Reduce faces when handling Big Data, this study encourages future Big Data research System architects must ensure that the data is protected and used only according to regulations. The scope of this document is on how Big Data can improve information security best practices. Identifying the best practices in Big Data privacy and increasing awareness of the threat to private information Machine learning as a data science to uncover patterns and hidden insights is not entirely a new concept. It has been in play with the use of neural networks starting in the 1980's. The question therefore is, "Why is there a big buzz around machine learning today?" The answer deception in the fact that development in technology and science has enabled game-changing differences in how machine learning algorithms have evolved and being applied. For example, traditionally, human-generated rule sets were the most prevalent approach in fraud management and still continue to be in practice today. But the required leap in computing power and accessibility of big data over the last five years has disrupted how data is being used to identify and prevent fraud. In the Big Data community, Map Reduce has been seen as one of the key enabling approaches for meeting continuously increasing demands on computing resources imposed by massive data sets. The reason for this is the high capacity of the Map Reduce model which allows for extremely equivalent and disseminated execution over a large number of computing nodes. This paper identifies Map Reduce issues and challenges in handling Big Data with the objective of providing an overview of the field, facilitating better planning and management of Big Data projects, and finding opportunities for future research in this field. The identified challenges are grouped into four main types corresponding to Big Data tasks types: online processing, data storage, security and privacy Machine learning uses artificially intelligent computer systems to separately learn, predict, act and explain without being clearly programmed. Simply put, machine learning eliminates the use of preprogrammed rule sets no matter how complex.**

Index Terms- **Big Data Analytics, Machine learning Big Data, MapReduce, NoSQL, Interactive Analytics, Privacy, Security**

I. INTRODUCTION

Big data—by which I mean the use of machine learning, statistical analysis, and other data mining techniques to extract hidden information and surprising correlations from very large and diverse data sets—raises numerous privacy concerns. The term Big Data refers to large-scale information management and analysis technologies that exceed the capability of traditional data processing technologies.1 Big Data is differentiated from traditional technologies in three ways: the amount of data (volume), the rate of data generation and transmission (velocity), and the types of structured and unstructured data (variety) The term "Big Data" refers to large and complex data sets made up of a variety of structured and unstructured data which are too big, too fast, or too hard to be managed by traditional techniques. Big Data is described by the 4Vs [2]: volume, velocity, variety, and veracity.

Volume refers to the quantity of data, variety refers to the diversity of data types, velocity refers both to how fast data are generated and how fast they must be processed, and veracity is the ability to trust the data to be accurate and reliable when making crucial decisions. Enterprises are aware that Big Data has the potential to impact core business processes, provide competitive advantage, and increase revenues [2]. Thus, organizations are exploring ways to make better use of Big Data by analyzing them to find meaningful insights which would lead to better business decisions and add value to their business. Map Reduce is a extremely scalable programming standard capable of dispensation massive volumes of data by means of parallel execution on a large number of product computing nodes. It was recently popularized by Google [3], but today the Map Reduce paradigm has been implemented in many open source projects, the most prominent being the Apache Hadoop [4]. The attractiveness of Map Reduce can be accredited to its high scalability, fault-tolerance, cleanness and independence from the program language or the data storage system. In the Big Data community, Map Reduce has been seen as one of the key enabling approaches for meeting the always increasing demands on computing resources imposed by massive data sets. on the related time, Map Reduce faces a number of obstacles when production with Big Data including the lack of a high-level language such as SQL, challenges in implementing iterative algorithms, support for iterative ad-hoc data looking at, and torrent processing. This paper aims to recognize issues and challenges faced by Map Reduce when confronted by Big Data with the objectives of: a) providing an overview an categorization of the Map Reduce issues and challenges, b) facilitating better planning and administration of Big Data projects and c) identifying opportunities for future research in this field. Map Reduce-related concepts have been previously presented and published a assessment of approaches focused on the support of circulated data management and processing using MapReduce.[5] They discussed implementations of database operators in MapReduce and DBMS implementations using MapReduce, while this paper is concerned with identifying MapReduce challenges in Big Data. Surveyed approaches to data dispensation based on the MapReduce model. Moreover, they analyzed systems which provide declarative programming interfaces on top of MapReduce. This paper aims to recognize challenges that MapReduce faces conduct Big Data. Moreover, this paper discusses security and privacy issues. The identified MapReduce challenges are grouped into four main categories corresponding to Big Data tasks types: data storage, analytics, online processing, security and privacy. in addition, this paper presents current efforts aimed at improving and extending MapReduce to address the identified challenges. Human beings now create 2.5 quintillion bytes of data per day. The rate of data creation has increased so much that 90% of the data in the world today has been created in the last two years alone. This acceleration in the production of information has created a need for new technologies to study massive data sets.[2] The importance for mutual research on Big Data topics is underscored by the U.S. federal government's recent $200 million funding initiative to carry Big Data research. To protect the data how Big Data is changeable security analytics by given that new techniques and opportunities for leveraging huge quantities of structured and unstructured data. The remainder of this document is organized as follows: Section 1 highlights the differences between conventional analytics and Big Data analytics, and briefly discusses tools used in Big Data analytics. The Big Data analytics on security and other describe to provides examples of Big Data usage in defense contexts. Section 5 describes a platform for testing on anti-virus telemetry data. Finally, Section 6 proposes a series of open questions about the role of Big Data in security analytics. some tools can help analysts generate complex queries and run machine learning algorithms on top of Hadoop. These tools include Pig (a platform and a scripting language for complex queries), Hive (an SQL-friendly query language), and Mahout and RHadoop (data mining and machine learning algorithms for Hadoop). New frameworks such as Spark4 were designed to advance the competence of data mining and machine learning algorithms that repeatedly reuse a working set of data, thus improving the efficiency of advanced data analytics algorithms. To complement the skills and capacities of human analysts, organizations are turning to Machine learning (ML) in hopes of providing a more forceful prevention. Quality Based Security method forecasts that "machine learning in virtual security will increase big data, intelligence, and analytics spending to $96 billion by 2021." At the SEI, machine learning has played a critical role across several technologies and practices that we proposed to develop to reduce the opportunity for and limit the damage of cyber attacks. In this post--the first in a series importance the application of machine learning Machine learning refers to systems that are able to by design progress with experience. Conventionally, no matter how many times you use software to perform the same exact task, the software won't get any smarter. Always launch your browser and visit the same exact website? A conventional browser won't "learn" that it should probably just bring you there by itself when first launched. With Machine Learning, software can gain the talent to learn from earlier explanation to make inferences about both future behavior, as well as guess what you want to do in new scenarios, Machine Learning has found a niche in all aspects of our daily life. To understand how Machine Learning works we first need to understand the fuel that makes Machine Learning possible: data. Consider an email spam detection algorithm. Original spam filters would simply blacklist certain addresses and allow other mail through. Machine learning enhanced this considerably by comparing verified spam emails with established rightful email and seeing which "features" were present more regularly in one or the other. For example, purposely misspelled words ("V!AGR4"), the presence of hyperlinks to recognized malicious websites, and virus-laden attachments are likely features analytical of spam rather than rightful email. (More discussion on "features" below.) This process of repeatedly inferring a label (i.e., "spam" vs "legitimate") is called classification, and is one of the major applications of Machine Learning techniques. It is worth mentioning that one other very common technique is forecasting, the use of chronological data to calculate future behavior. While substantial research and knowledge has been developed to achieve forecasting, the remainder of this post will focus on classification.

II. RELATED WORK

2.1 INFRASTRUCTURE OF BIG DATA AND MACHINE LEARNING

To handle different magnitude of big data in terms of volume, velocity, and variety, we need to design efficient and effective systems to process large amount of data arriving at very high speed from different sources. Big data has to go through multiple phases during its life cycle. Data are circulated nowadays and new technologies are being developed to store and process large repositories of data. For example, cloud computing technologies, such as Hadoop MapReduce, are explored for big data storage and processing. In this section we will explain the life cycle of big data and Machine learning. In addition, we will also discuss how big data are leveraging from cloud computing technologies and drawbacks associated with cloud computing when used for storage and processing of big data.

Data creation: Data can be created from various distributed sources. The amount of data generated by humans and machines has exploded in the past few years. For example, everyday 2.5 quintillion bytes of data are generated on the web and 90 percent of the data in the world is generated in the past few years. Face book, a social networking site alone is generating 25TB of new data every day. Frequently, the data generated is large, various and complex. Therefore, it is hard for traditional systems to handle them. The data generated are normally associated with a specific domain such as business, Internet, research, etc. Data storage: This phase refers to storing and managing large-scale data sets. A data storage system consists of two parts i.e., hardware communications and data management [6]. Hardware communications refers to utilizing information and communications technology (ICT) resources for various tasks (such as distributed storage). Data management refers to the set of software deployed on top of hardware infrastructure to manage and query large scale data sets. It should also provide several interfaces to interact with and analyze stored data. ²

Data processing: Data processing phase refers basically to the process of data gathering, data broadcast, preprocessing and extracting useful information. Data collection is needed because data may be coming from different diverse sources i.e., sites that contains text, images and videos. In data collection phase, data are acquired from specific data construction environment using dedicated data collection technology. In data transmission phase to define the steps needed for any model-based machine learning application are:

1. Gather data for training and evaluating the model.
2. Gather knowledge to help make appropriate modeling assumptions.
3. Visualize the data to understand it better, check for data issues and gain insight into useful modeling assumptions.
4. Construct a model which captures knowledge about the problem domain, consistent with your understanding of the data.
5. Perform inference to make predictions over the variables of interest using the data to fix the values of other variables.
6. Evaluate results using some evaluation metric, to see if they meet the success criteria for the target application. In the (usual) case that the system does not meet the success criteria the first time around, there are then two additional steps needed.
7. Diagnose issues which are reducing prediction accuracy. Visualizations is a commanding tool for bringing to light troubles with data, models or inference algorithms. Inference issues can also be diagnosed using synthetic data sampled from the model .At this stage, is may also be necessary to diagnose presentation issues if the inference algorithm is taking too long to complete.
8. Refine the system – this could mean refinement the data, model, visualizations, inference or valuation.

III. BIG DATA ANALYTICS

3.1Machine Learning

Machine Learning is an area of Computer Science which provides Computers to learn themselves and change them when performance to new data without being clearly planned. Forecast is the final goal of Machine Learning. Machine Learning-based IDs provide a good presentation and require less professional knowledge. It used supervised learning techniques to classify intrusions. Categorization algorithms include Decision Trees and its derivatives like c4.5, c5.0, Naïve Bayes, K-Nearest Neighbour (KNN), Hidden Markov Model, Logistic regression and many more. This paper studies two of these algorithms which are Naïve Bayes and KNN. The incidence and attractiveness of Big Data offers the assure of building more intelligent decision making systems. This is because the typical basis for many decision making algorithms is that more data can better teach the algorithms to create more accurate outputs. The key to extracting useful information from Big Data lies within the use of Machine Learning (ML) approaches. However, the use of huge datasets themselves for the purpose of analysis and training poses some problems and challenges to the very effective of Machine Learning algorithms. The arithmetic and computational complexity brought on by the volume constituent of Big Data renders traditional Machine Learning (ML) algorithms almost unusable in usual development environments. This is due to the fact that Machine Learning (ML) algorithms were designed to be used on much smaller dataset with the statement that the entire data could be held in memory [9]. With the arrival of Big Data, this premise is no longer valid and therefore really impedes the custom of those algorithms. In order to remediate to this problem, circulated processing algorithms such as MapReduce were brought forward. Although some ML algorithms are essentially parallel and can therefore be adapted to the MapReduce paradigm [10], for others the transition is much more complex. The foundation of many ML algorithms relies on strategies directly dependent on in-memory data and therefore once that statement is detached, entire families of algorithms are rendered insufficient. The parallel and distributive nature of the MapReduce model is a source of such a disconnect. This is what Parker [10] describes as the curse of modularity. The subsequent families of algorithms are :

□ Gradient Descent algorithms: The chronological nature of these algorithms requires a very large amount of jobs to be chained. It also requires that parameters be updated after each iteration, which will add communication overhead to the process. Both of these steps are therefore expensive in terms of time.

- Expectation Maximization algorithms: equally this family of algorithm also depends on iterations that are implemented as jobs, causing the same routine latencies as above. In order to address the shortcomings of MapReduce, alternatives have been developed to function either separately or in addition to accessible MapReduce implementations [11].

- Iterative Graph algorithms: Much iteration are necessary in order to reach meeting, each of which corresponds to a job in MapReduce [11] and jobs are costly in terms of startup time. in addition, skews in the data create stragglers in the Reduce phase, which causes backup execution to be launched, growing the computational load [3].
- Hadoop [12] and Twister [13] are both extensions designed for Hadoop [3] in order for this MapReduce implementation to enhance sustain iterative algorithms. Each of these tools possesses its strengths and area of focus but the difficult integration and possible incompatibilities between the tools and frameworks reveal new research opportunities that would fulfill the need for a uniform ML solution. When taking into consideration the amount constituent of Big Data, extra arithmetical and computational challenges are revealed. Regardless of the paradigm used to develop the algorithms, an important determinant of the success of supervised ML approaches is the pre-processing of the data. This step is often significant in arrange to obtaining dependable and significant results. Data cleaning, normalization, feature extraction and selection [14] are all essential in order to obtain an appropriate training set. This poses a enormous challenge in the light of Big Data as the preprocessing of enormous amounts of tuples is often not possible.

In particular, the conception of sound has provided a paradigm shift in the underlying algebra used for ML algorithms. Dalessandro [15] illustrates the usefulness of accepting noise as a given, and then using more efficient, but less accurate, learning models. Dalessandro shows that using computationally less costly algorithms, which are also less correct during middle steps, will define a model which performs equally well in predicting new outputs when trained on Big Data. These algorithms may take more iterations than their computationally more expensive counterparts; however, the iterations are much faster. Due to this, the less expensive algorithms tend to converge much faster, while giving the same accuracy. An example of such an algorithm is stochastic incline descent [15].

In addition to the challenges mentioned above, having a variety of dissimilar data sources, each storing dissimilar data types, can also affect the show of the ML algorithms. Information preprocessing strength get enhanced a few of those challenges and is largely significant in the Map Reduce model where outliers can greatly control the performance of algorithms [16]. In order to remediate to these problems, solutions have been developed to implement data preprocessing algorithms using Map Reduce [17]. However, it is still necessary to find ways to integrate the analysis and preprocessing phase, which create new research scenario. The velocity component of Big Data introduces the idea of concept drift within the learning model.

In Map Reduce, this idea is motivated by the requirement to pre-process data, which introduces extra delays. The fast coming of data along with potentially long computing time may cause a concept drift, which Yang and Fong define as “known problem in data analytics, in which the statistical properties of the attributes and their target classes shift over time, making the trained model less accurate”[18]. Thus accurate concept drift detection constitutes an important research area to insure accuracy of ML approaches with Big Data. An important subset of ML algorithms is analytical modeling. That is, given a set of known inputs and outputs, can we predict an unknown output with some probability. For example, to predict movies that clients will enjoy, companies such as Yahoo and Netflix collect a large variety of information on their clients to build accurate recommender systems. From the author’s inspection, parallelism techniques for analytical modeling fall into three categories of implementation:

1. Run the analytical algorithm on subsets of the data, and return all the results.
2. Generate intermediate results from subsets of the data, and resolve the intermediate results into a final result.
3. Parallelize the underlying linear algebra.

The two most talented forms of implementation for Big Data are type 2 and 3. type 2 is basically the definition of a MapReduce job; where the algorithm attempts to produce middle results using Map operations, and combines these outputs using Reduce operations. type 3 can also be seen as a MapReduce job, if the underlying linear algebra separable into

Map and Reduce operations. Finally, type 1 is basically not a valid clarification for Big Data as the results are only analytical of small subsets of the data and not the forecast over the entire dataset. MapReduce with predictive modeling has a major constraint which limits its helpfulness when predicting extremely associated data. MapReduce works well in contexts where clarification can be processed individually. In this case the data can be split up, designed, and then aggregated together. but, if there are associated clarification that need to be processed together, MapReduce offers little benefit over non distributed architectures. This is because it will be fairly general that the explanations that are associated are found within disparate clusters, leading to large performance overheads for data communication between clusters. Use cases such as this are commonly found in predicting stock market fluctuations. There are a few possible solutions based on solutions from predictive modeling on traditional data sizes: data reduction, data aggregation, and sampling. To handle to agree to MapReduce to be used in these types of logical modeling problems [19].

3.2. Interactive Analytics

An interactive analytics can be distinct as a set of approaches to agree to data scientists to learn data in an interactive way, supporting evaluation at the rate of individual thought [20]. Interactive analytics on Big Data provides some stimulating research areas and distinctive problems. The important difference to other data analytic paradigms is the idea of interactive rates. By definition, interactive analysis requires the user to frequently adjust their approach to produce inspiring analytics [21]. MapReduce for interactive analytics poses a great shift from the classic MapReduce use case of dispensation batch computations. Interactive analytics involves performing several small, short, and interactive jobs. As interactive analytics starts to move from RDBMSs to Big Data storage systems some preceding assumptions relating to Map Reduce are broken, such as standard data right of entry and event of large batch jobs. This type of analysis requires a new class of MapReduce workloads to deal with the interactive, almost real-time data models [22] discuss these considerations in their study of developed methods where the authors discover that extending Map Reduce with querying frameworks such as Pig and Hive are prevalent. note that interactive analysis for Big Data can be seen as an expansion of the already well-researched area of interactive query processing. Production this statement, there exist possible solutions to optimize interactive analytics with MapReduce by mirroring the already existing work in interactive query processing. One open area of potential work is finding the best method to bring these solutions to the MapReduce programming model. MapReduce is one parallelism model for interactive analytics. an additional approach tuned for interactivity is Google's Dremel system [23], which acts in complement to MapReduce. Dremel builds on a novel column-family storage format, as well as algorithms that constructs the columns and reassemble the original data. Some highlights of the Dremel system are:

- Close to linear scalability in the number of clusters.
- Premature termination, similar to progressive analytics, to provide speed tradeoffs for accuracy.
- synchronized interactivity for scan-based queries

Other interactive analytics research has been based on the column-family NoSQL data storage approach. The main benefit of column-based approaches versus rowbased, traditional, approaches is that only a fraction of the data needs to be accessed when processing typical queries [7]. However, most of these approaches are specialized for certain types of datasets and certain queries and thus provide an open research area for a generalized solution.

3.2.1.Data Visualization

A large category of interactive analytics is data visualization. There are two primary problems associated with Big Data visualization. First, many instances of Big Data involve datasets with large amount of features, wide datasets, and building a highly multi-dimensional visualization is a difficult task. next, as data grows better vertically, tall datasets, uninformative visualizations are usually produced. For these tall datasets, the resolution of the data must be limited, i.e. through a process to collective outputs to ensure that highly dense data can still be deciphered [20]. For highly wide datasets, a preprocessing step to reduce the dimensionality is needed. regrettably this tends to be useful on tens to hundreds of dimensions, for even higher dimensions a mixed-initiative method, including human involvement, to determine subsets of related dimensions is required [20]. This move toward usually requires individual contribution to determine an initial subset of "interesting" features, which is also a difficult task and open research area. MapReduce for data revelation presently performs well in two cases: memory-insensitive visualization algorithms, and inherently parallel visualization algorithms. [24] Have provided a study on moving existing visualization algorithms to the MapReduce paradigm. One major input is empirically proving that MapReduce provides a high-quality solution to major tentative visualization. The authors present that this is because scalability is achieved through information reduction tasks which can be highly parallel; these types of tasks are common in data visualization algorithms. Further, visualization algorithms that tend to increase the total amount of data for middle steps will perform poorly when mapping to the MapReduce paradigm. Another drawback to MapReduce with visualization is that a typical MapReduce job uses one pass over the data. Therefore, algorithms that need several iterations, such as mesh simplification, will suffer from a large overhead in trying to naively map the algorithm to the MapReduce paradigm. This is similar to the problems created for iterative machine learning algorithms. Therefore, there is the potential for research aimed at providing optimized multiple iteration solutions for MapReduce.

IV. ONLINE PROCESSING

The speed measurement, as one of the Vs used to define Big Data, brings many new challenges to conventional data processing approaches and particularly to MapReduce. Handling of Big Data often requires applications with online processing capabilities, which can be broadly defined as real-time or quasi real-time processing of fast and continuously generated data (also known as data streams). From the business perspective, the goal is normally to obtain insights from these data streams, and to enable prompt reaction to them. This instantaneous reaction can bring business value and competitive advantage to organizations, and therefore has been generating research and commercial interest. Areas such as economic fraud finding and algorithmic trading have been highly interested in this type of solutions. The MapReduce paradigm is not an appropriate solution for this kind of low-latency processing because:

- MapReduce computations are batch processes that start and finish, while computations over streams are continuous tasks that only finish upon user request.
- Some use cases require a response time that is very difficult to achieve in networked environments.
- In order to provide fault tolerance, most of MapReduce implementations, such as Google's [4] and Hadoop [3],write the results of the Map phase to local files before sending them to the reducers. In adding, these implementations store the output files in distributed and high-overhead file systems (Google File System [26] or HDFS [4], respectively). File handling adds important latency to the processing pipelines.
- The inputs of MapReduce computations are snapshots of data stored on files, and the content of these files do not change during processing. Equally, data streams are incessantly generated and uncontrolled inputs [25].
- Not all computation can be professionally expressed using the MapReduce programming model, and the model does not natively support the composition of jobs. Minute batches and all dispensation is performed on these batches, which difference with the event-by-event dispensation in Storm and S4. Despite all the advancements described in this section, there still are many challenges related to online processing of Big Data, such as:
- There is no high-level standardized language that can be used to express online computations.

V. PRIVACY AND SECURITY

In this section privacy and security and concerns for MapReduce and Big Data are discussed. Also, current efforts to address these problems for MapReduce are accessible. responsibility and auditing are security issues that present a problem for both MapReduce and Big Data. responsibility is the ability to know when someone performs an action and to hold them accountable for that action and is often tracked through auditing. In MapReduce accountability is only provided when the mappers and reducers are held responsible for the tasks they have completed [27]. Single clarification to this subject that has been proposed is the creation of an Accountable MapReduce [27]. This solution utilizes a set of auditors to inconspicuously perform accountability tests on the mappers and reducers in real-time [27]. during the monitoring of the results of these tests, spiteful mappers or reducers can be detected and responsibility can be provided. An additional security challenge presented to MapReduce and Big Data is that of providing access control, which can be shown through three of Big Data's defining V properties: volume, variety and velocity [28].

- Both require in their respective jurisdictions that individuals who have data collected on them are able to understand how it is being used, by whom, and for what purposes. Abiding by such legislation is difficult for any large data environment.
- Both state that in some circumstances consent must be given before information can be used. Due to the size of the data and the complexity of the analytics performed during a MapReduce, informing an individual about what is happening to their information is a challenge.
- Both state that consent can be withdrawn and if so the information should be deleted by the data repository. However, in Big Data once information has been put into the system it is difficult if not impossible to remove. several work has been done in order to give privacy protection for MapReduce

While production with a large volume of information, work performed on that information is likely to require access to several storage locations and devices. Consequently, several access requirements will be essential for any one task. When dealing with data that has a large variety, semantic considerate of the data should play a role in the access control decision process [28]. Finally, the velocity requirement of MapReduce and Big Data requires that whatever access control approach is used must be optimized to determine access control rights in a reasonable amount of time. Privacy is a major topic of concern whenever large amounts of information are used. Processes such as data mining and predictive analytics can discover or deduce information linkages. in order linkages are beneficial to organizations, allowing them to better understand, target and provide for their clients or users. However, on an individual basis this discovery of information can cause the identities of data providers to be exposed. Privacy protection requires an individual to maintain a level of control over their personal. User input allows an individual to state their private information usage wishes. Transparency is provided to an individual by the knowledge of how private information is collected, what private information is collected, how the private information is being used, and who has access to it. This can be very complex when dealing with a large number of mappers and reducers that MapReduce often requires. It is possible that the ability to provide simplicity and control is stated in legislation that must be followed or penalties can be incurred. The following are examples of issues that could lead to penalties using the example of the Personal Information Protection and Electronic Documents Act (PIPEDA) in Canada [29], and the Data Protection Directive of the European Union [30].

VI. CONCLUSIONS

Conventional data processing and storage approaches are facing many challenges in gathering the continuously increasing computing demands of Big Data. This work focused on MapReduce, one of the key enabling approaches for meeting Big Data demands by means of highly parallel processing on a large number of product nodes. Issues and challenges MapReduce faces when dealing with Big Data are identified and categorized according to four main Big Data task types: data storage, analytics, online processing, and security and privacy. Additionally, efforts aimed at improving and extending MapReduce to address identified challenges are presented. By identifying MapReduce challenges in Big Data, this paper provides an overview of the field, facilitates better planning of Big Data projects and identifies opportunities for future research.

REFERENCES

[1] P. Zadrozny and R. Kodali, Big Data Analytics using Splunk, Berkeley, CA, USA: Apress, 2013.

[2] F. Ohlhorst, Big Data Analytics: Turning Big Data into Big Money, Hoboken, N.J, USA: Wiley, 2013.

[3] Apache Hadoop, http://hadoop.apache.org.

[4] J. Dean and S. Ghemawat, "MapReduce: Simplified data processing on large clusters," Commun ACM, 51(1), pp. 107-113, 2008.

[5] F. Li, B. C. Ooi, M. T. Özsu and S. Wu, "Distributed datamanagement using MapReduce," ACM Computing Surveys, 46(3), pp. 1-42, 2014.

[6] C. Doulkeridis and K. Nørvåg, "A survey of large-scale analytical query processing in MapReduce," The VLDB Journal, pp. 1-26, 2013.

[7] K. Grolinger, W. A. Higashino, A. Tiwari and M. A. Capretz, "Data management in cloud environments: NoSQL and NewSQL data stores," Journal of Cloud Computing: Advances, Systems and Application, 2, 2013.

[8] K. A. Kumar, J. Gluck, A. Deshpande and J. Lin, "Hone: Scaling down hadoop on shared-memory systems," Proc. of the VLDB Endowment, 6(12), pp. 1354-1357, 2013.

[9] C. Parker, "Unexpected challenges in large scale machine learning," Proc. of the 1st International Workshop on Big Data, Streams and Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications, 2012.

[10] J. Lin, "Mapreduce is good enough? if all you have is a hammer, throw away everything that's not a nail!" Big Data, 1(1), pp. 28-37, 2013.

[11] Y. Bu, B. Howe, M. Balazinska and M. D. Ernst, "HaLoop: Efficient iterative data processing on large clusters," Proc.VLDB Endow., 3(1- 2), pp. 285-296, 2010.

[12] J. Ekanayake, H. Li, B. Zhang, T. Gunarathne, S. Bae, J. Qiu and G.Fox, "Twister: A runtime for iterative MapReduce," Proc. of the 19th ACM International Symposium on High Performance Distributed Computing, 2010.

[13] S. B. Kotsiantis, D. Kanellopoulos and P. Pintelas, "Data preprocessing for supervised learning," International Journal of Computer Science, 1(2), pp. 111, 2006.

[14] B. Dalessandro, "Bring the noise: Embracing randomness is the key to scaling up machine learning algorithms," Big Data, 1(2), pp. 110-112, 2013.

[15] G. Ananthanarayanan, S. Kandula, A. Greenberg, I. Stoica, Y. Lu, B. Saha and E. Harris, "Reining in the outliers in map-reduce clusters using Mantri," Proc. of the 9th USENIX Conference on Operating Systems Design and Implementation, 2010.

[16] Q. He, Q. Tan, X. Ma and Z. Shi, "The high-activity parallel implementatio n of data preprocessing based on MapReduce," Proc. Of the 5th International Conference on Rough Set and Knowledge Technology, 2010.

[17] H. Yang and S. Fong, "Countering the concept-drift problem in Big Data using iOVFDT," IEEE International Congress on Big Data, 2013.

[18] T. Hill and P. Lewicki, STATISTICS: Methods and Applications, Tulsa, OK: StatSoft, 2007.

[19] J. Heer and S. Kandel, "Interactive analysis of Big Data," XRDS: Crossroads, the ACM Magazine for Students, 19(1), pp. 50-54, 2012.

[20] P. Bhatotia, A. Wieder, R. Rodrigues, U. A. Acar and R. Pasquin,"Incoop: MapReduce for incremental computations," Proc. of the 2nd ACM Symposium on Cloud Computing, 2011.

[21] Y. Chen, S. Alspaugh and R. Katz, "Interactive analytical processing in Big Data systems: A cross-industry study of MapReduce workloads," Proc. of the VLDB Endowment, 5(12), pp. 1802-1813, 2012.

[22] S. Melnik, A. Gubarev, J. J. Long, G. Romer, S. Shivakumar, M. Tolton and T. Vassilakis, "Dremel: Interactive analysis of Web-scale datasets," Proc. of the VLDB Endowment, 3(1-2), pp. 330-339, 2010.

[23] H. T. Vo, J. Bronson, B. Summa, J. L. Comba, J. Freire, B. Howe, V. Pascucci and C. T. Silva, "Parallel visualization on large clusters using MapReduce," IEEE Symposium on Large Data Analysis and Visualization, 2011.

[24] W. Lam, L. Liu, S. Prasad, A. Rajaraman, Z. Vacheri and A. Doan, "Muppet: MapReduce-style processing of fast data," Proc.VLDB Endow., 5(12), pp. 1814-1825, 2012.

[25] S. Ghemawat, H. Gobioff and S. Leung, "The Google file system," ACM SIGOPS Operating Systems Review, 2003.

[26] Z. Xiao and Y. Xiao, "Achieving accountable MapReduce in cloud computing," Future Generation Computer Systems, 30,pp. 1-13, 2014.

[27] W. Zeng, Y. Yang and B. Luo, "Access control for Big Data using data content," IEEE International Conference on Big Data, 2013.

[28] The Personal Information Protection and Electronic Documents Act (PIPEDA), http://www.priv.gc.ca/leg_c/r_o_p_e.asp.

[29] Protection of Personal Data, http://ec.europa.eu/justice/dataprotection.