Hybrid Approach of Big Data File Classification Based on Threat Analysis for Enhancing Security

Saranya N
Department of Computer Science and Engineering,
Sri Eshwar College of Engineering, Coimbatore, India.
saranya.n@sece.ac.in

Article Info
A. Haldorai et al. (eds.), *2nd International Conference on Materials Science and Sustainable Manufacturing Technology*, Advances in Computational Intelligence in Materials Science.
Doi: https://doi.org/10.53759/acims/978-9914-9946-9-8_24

Abstract – Big Data is rapidly growing domain across various real time areas like Banking, Finance, Indusrty, Medicine, Trading and so on. Due to its diversified application, handling the big data for security during data transmission or management is highly risky. Most of the researchers try to handle big data classification based on the domain of interest for increasing productivity or customer satisfaction in decision making. Whereas, this paper focuses on the classification of big data file to enhance security during the data transmission over network and management.Most of the big data applications contains valuable and confidential data. The existing data security approaches are not sufficient on handling the security for data based on the threat level. Therefore, this paper proposes a hybrid approach to classify the big data based on the threat level of the contents associated with the data under consideration into open and close. To ensure the security of big data files, they are transmitted into the Hadoop Distributed File System along with relevant information to assess the level of threat they pose. The Threat Impact Level (TIL) is then calculated as a metric to determine the threshold level required for their protection.

Keywords – Data Security, Big Data, Hadoop Distributed File System, Classification, Threat Impact Level.

I. INTRODUCTION

In digital era, Big Data is the rapidly growing domain in terms of its diversified nature and data size. Big data is a huge voluminous data growing exponentially in time [1]. The four major V's supported by Big Data are Volume, Variety, Velocity and Veracity as depicted in **Fig 1.** Volume is related to the size of the data. Variety is related to the heterogenous sources of data.Velocity refers to the speed of creation of data.Veracity referes to the inconsistency of the data which is also referred as Variability.

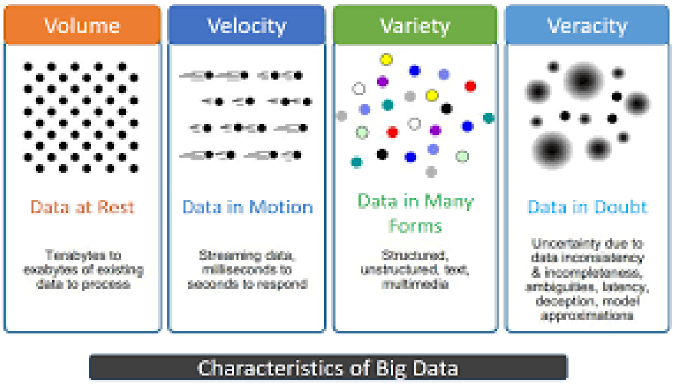
Volume
Velocity
Variety
Veracity
Data at Rest
Data in Motion
Data in Many Forms
Data in Doubt
Terabytes to exabytes of existing data to process
Streaming data, milliseconds to seconds to respond
Structured, unstructured, text, multimedia
Uncertainty due to data inconsistency & incompleteness, ambiguities, latency, deception, model approximations
Characteristics of Big Data

Fig 1. Characteristics of Big Data

Big data can be classified into three main types: structured, unstructured, and semi-structured. Structured data is typically stored, managed, and retrieved in a fixed format, often generated from relational databases. In contrast, unstructured data is

.

disorganized and collected from various sources, such as search engines. Semi-structured data combines features of both structured and unstructured data and is often analyzed in file formats like XML.

When it comes to processing and analyzing big data, security and privacy are critical concerns. Various approaches have been proposed to ensure the confidentiality, integrity, and availability of big data, as discussed in reference [2].

Hadoop Distributed File System (HDFS) is a Big Data analysis tool designed to manage and store massive data sets while automating high-speed evaluations and minimizing the need for manual assessments. HDFS is widely recognized for its reliability, scalability, support for distributed architecture systems, and parallel processing capabilities. It can effectively manage various types of Big Data, including structured, semi-structured, and unstructured data.

The Hadoop MapReduce framework is a distributed computing model that provides a reliable and efficient method for processing large datasets on large clusters. Its architecture is tailored to handle big data, and it comprises two main tasks, namely Map and Reduce.The Hadoop MapReduce framework follows a two-stage process: the Map task and the Reduce task. During the Map task, input data is transformed into an output set of data by breaking down its components into tuples and forming key-value pairs. The Reduce task takes the output of the Map task as input and merges data tuples into a reduced set of tuples. It is noteworthy that the Reduce task always follows the Map job. A visual representation of this process is provided in **Fig 2.**

In a distributed network environment, utilizing the Hadoop MapReduce Job-Scheduling algorithm can aid in clustering Big Data and leveraging Hadoop technologies and HDFS tools can provide solutions for various information security challenges related to Big Data analysis. [7].

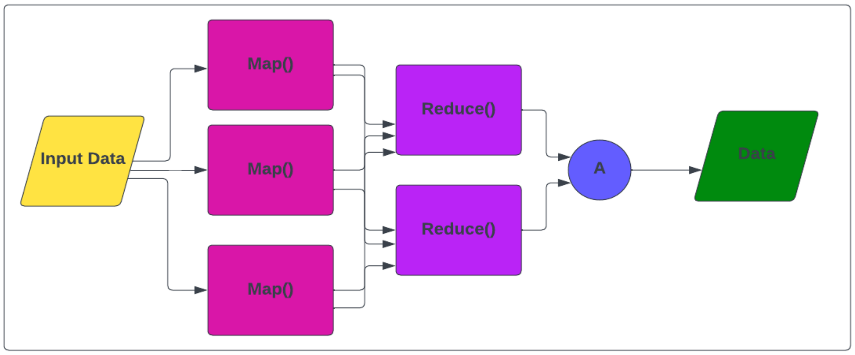

Fig 2. Map-Reduce Framework

The data utilized in big data analysis is critical and presents numerous challenges for secure handling. As a result, researchers aim to reduce potential risks associated with handling sensitive data, particularly since much of the data is derived from personal, financial, medical, and industrial sources. The protection of Big Data from cyber threats is a major challenge, and cybersecurity assumes great importance in this regard. Big Data is highly coveted by malicious actors who seek to access valuable information. Due to the size and complexity of big data, conventional security measures may not be effective in safeguarding the data from potential threats. As a result, new and innovative technologies are being developed to provide enhanced protection for big data against security risks [8-10].

When it comes to storing, managing, analyzing, and transferring Big Data, security and privacy are two of the most important considerations to keep in mind [10]. When creating Big Data environments and ensuring the security of stored or processed data, various encryption methods are employed to prevent unauthorized access by third parties [11]-[12], while also addressing privacy concerns. However, some of the proposed encryption methods are impractical due to their time complexities, despite giving equal priority to all data. Two key aspects of Big Data security are data security and access control [13], with data management, storage, and classification emerging as the most pressing issues [14].

The main contribution of this paper is the classification of big data contents to identify the threat level for data transmission or storage in future. The proposed work makes three key contributions.

- A new classification method for Big Data is introduced, which evaluates data sensitivity and criticality based on data content and metadata characteristics.
- This approach addresses the challenge of processing vast amounts of data by adapting the HDFS formatter, an open-source distributed software platform, for high-speed data classification.
- The files that make up Big Data are separated into two distinct groups: public files, which do not need to be protected, and confidential files, which do. These categories are determined by the predefined policies that are established by the owner of the data, who establishes the appropriate level of confidentiality.

- As a consequence, just the essential data that needs to be protected is encrypted, which lowers the cost of maintaining security for files that are transferred between cloud nodes..

The paper is structured into several sections. Section II reviews the literature on Big Data classification, while Section III presents the proposed approach for Big Data classification, which calculates threat levels to enhance security.Section IV provides the implementation details and algorithmic steps have explained and section V examines the results obtained from the proposed approach and discussions. Finally,the section VI provides the conclusion and future work of the proposed work in this paper.

II. LITERATURE SURVEY

This section contains a literature review in which various methods for big data classification in the literature are discussed, as well as their limits, as follows.

Classification is a vital data analysis tool for determining the attribute class. To reduce processing time and data volume, Isaac Triguero et al. [14] developed a clustering-based distributed nearest neighbor classification method. Map-reduce was used to implement the algorithm on a cluster of computing devices, thereby enhancing the clustering-based sampling procedure. Although the method achieved high reduction rates without sacrificing precision, it did not adequately address the dimensionality issue.

Isaac Triguero et al. [14] proposed a clustering-based distributed nearest neighbor classification technique using Map-Reduce to reduce the number of data records processed and achieve high reduction rates while maintaining accuracy. However, the method was limited in addressing the dimensionality issue.

Simone Scardapane et al. [15] proposed the Echo State Networks algorithm for big data classification and relied on local communication between neighboring elements without requiring connected nodes or training patterns. Although the algorithm showed promising results on synthetic datasets, the aggregation of weights without considering error values was a limitation.

Another notable contribution was made by Jemal H. Abawajy et al. [16], who proposed the Large Iterative Multitier Ensemble (LIME) classifiers. These classifiers integrated different ensemble meta-classifiers at various levels, resulting in an iterative system capable of successfully classifying large amounts of data. However, the high computing cost of the ensemble classifiers was a drawback.

Junchang Xin et al. [17] developed the Elastic Extreme Learning Machine (E2LM), a distributed classifier using the Extreme Learning Machine (ELM) and MapReduce principle for effective learning from a huge training dataset. While this method required minimal human intervention and provided rapid training speed, the results needed improvement due to its limitations.

Alessio Bechini et al. [18] developed MapReduce-based Associative Classifier (MRAC), a distributed classification technique based on association rules and the MapReduce programming model. It used the FP-Growth algorithm to mine Classification Association Rules (CARs) and performed distributed rule pruning to classify unlabeled patterns.MRAC was able to achieve scalability and speed, but it was unable to run the experiment on a greater number of datasets.

In their study, Shichao Zhang et al. [19] utilized the k-Nearest Neighbours (kNN) technique to partition a large dataset into multiple segments using k-means clustering. They subsequently applied kNN classification to each segment. The method was tested on both medical imaging data and big data, and the results were found to be accurate and efficient. However, selecting the appropriate k value was crucial, as increasing k could result in reduced classification accuracy.

Diego Marrón et al. [20] proposed a random feature function filter that incorporated Hoeffding-Trees (HT), k-Nearest Neighbors (kNN), and Stochastic Gradient Descent (SGD) techniques to enhance predictive capability. The method yielded improved performance without extensive parameter tuning. However, optimization of memory efficiency and adaptability to the drifting concept can still be enhanced.

Anushree Priyadarshini and Sonali Agarwal [21] developed a MapReduce strategy utilizing Support Vector Machine (SVM) for large-scale data handling. The authors evaluated the SVM's efficiency by studying the impact of penalty and kernel parameters. While computing time was reduced, accuracy could still be affected negatively if SVM parameters were not optimized using an optimization method.

The GWO meta-heuristic algorithm, developed by SeyedaliMirjalili et al. [22], was inspired by the hunting strategy and leadership hierarchy of grey wolves. The researchers compared the performance of this algorithm to other meta-heuristics such as Evolution Strategy, Evolutionary Programming, Differential Evolution, Gravitational Search Algorithm, and Particle Swarm Optimization using 29 widely-used test functions. The findings indicated that the GWO algorithm was superior in terms of performance.

The protection of valuable information in big data is an essential objective to prevent any potential hazards. Big data analysis has become a crucial topic [23] in many industries, and data security is a significant concern. In fact, it is a complex problem that has been demonstrated to be NP-Hard [24]. In this context, confidentiality, integrity, and availability are the three primary considerations in big data analysis. Confidentiality ensures that sensitive data is accessible only to authorized users, based on its sensitivity. Integrity gives authorized users the ability to change, edit, update, or delete data, while availability ensures that information can be accessed quickly and easily whenever it is required.

However, many companies typically store, gather, and process a considerable amount of sensitive data in one location. This data may include personal information of customers and patients, financial and trading data, and other sensitive information. The risk of data leakage, data loss, hacking, and sabotage is increased when a large amount of confidential information is stored in a single location. A malicious attack may also lead to a denial of service, further disrupting business operations.

In order to address these potential risks, we suggest a novel method for classifying risk assessment that considers various metrics such as asset value, exposure to vulnerability, level of threat, and probability of a threat. The goal of this approach is to proactively manage risk and prevent potential threats to big data.By identifying the level of risk for each data asset, organizations can take necessary measures to ensure data security, such as implementing data encryption, access control, and intrusion detection systems.

In conclusion, big data security is a complex problem that requires careful consideration of various factors. Our proposed approach for risk assessment classification can aid in mitigating potential threats to big data, thereby promoting effective risk management practices.

The purpose of this study is to present an integrated classification and security approach with the primary attention being placed on the level of confidentiality maintained by the contents of files. The proposed approach includes a data security algorithm that specifically aims to mitigate the risks that come with large-scale data aggregation and migration.

III. PROPOSED APPROACH

The primary emphasis of this paper's proposed method is on classifying big data files to strengthen their security during transmission.The **Fig 3** depicts the schematic diagram of the Proposed Big Data File Classification Approach. The big data file to be transmissed with different level of security is considered and moved into the Hadoop Distributed File System. To begin with the Big Data Classification, the inputted file is converted into the text file which represents the input files format in HDFS environment. The format file is later splitted to n- number of big data fileby partitioning based on the file information. Now the next step is to map the file with MAP-REDUCE framework. Each partition of the file is mapped to the MAPPER and then to REDUCER. The Threat Impact Level of the file is computated from the output of the HDFS REDUCER. The classification level named Threat Impact Level is the threshold for enhancing the security to the file before transmission.

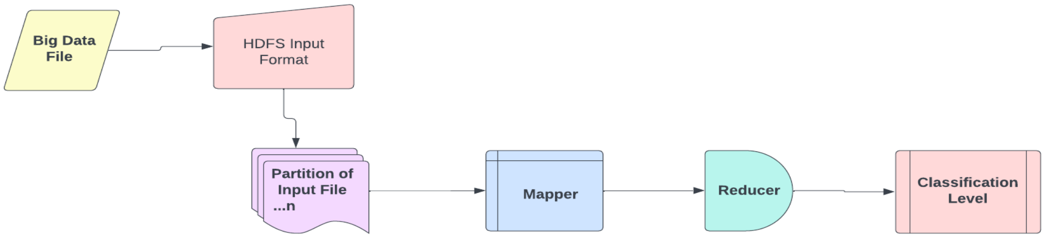

Fig 3. Schematic Diagram of proposed Big Data File Classification

Big Data File Classification Approach

Classifying big data according to its sensitivity, criticality, demands, priorities, and required level of security is possible. This categorization method divides big data into two categories based on the level of threat they pose: confidential data and public data.

Threat Impact Level (TIL) Computation

Each threat impact level is assigned a number between 0 and 5. The impact levels as listed in table 1 are based on ISO27005:2011.

Table 1. Threat Impact Levels

Impact Value	Level
0 – 1	Negligible
1 – 2	Low
2 – 3	Medium
3 – 4	High
4 – 5	Very High

In order to safeguard big data, the level of security control needed is determined by the Threat Impact Level (TIL) value. This value is calculated using threat metrics and indicates whether protection is necessary (critical) or not (public). Equation (1) is used to calculate the TIL value, which is the product of Asset, THV, and

$$\mathrm{LTH.TIL} = \mathrm{Asset}\ \times \mathrm{THV}\ \times \mathrm{LTH} \tag{1}$$

Where,
THV is the sum of the Threat and Vulnerability
LTH is the Likelihood of Threat

Extended Metadata Attribute (EMA) Computation

The meta data information is taken into account when computing the extended threat level of a big data file when computing the Threat Impact Level. Extended Metadata Attribute (EMA) is computed based on the TIL as in equation (2).

$$\mathrm{EMA} = \begin{cases} 0\,,\ 0 \leq \mathrm{TIL} \leq 1 \\ \ 1,\ \mathrm{TIL} > 1 \end{cases} \tag{2}$$

The EMA is considered to be True (1: confidential) when the threat impact level is between low and extremely high, while it is considered False (0: public) in all other circumstances. For the sake of classification, the EMA is appended to the metadata of the file whenever it is produced.

File Categories

Confidential data refers to highly sensitive information that can only be accessed by authorized personnel. This type of data includes critical business and client information, such as financial records, personal details, and sensitive medical data. Any unauthorized disclosure of such data can result in financial loss or harm the reputation of the organization. Secret metadata attributes are extra components of the file system that aren't evaluated by the system but offer supplementary data on files, including file permissions or modifications. System administrators can utilize these characteristics to add more metadata to a file or directory, which helps to protect them against potential security breaches.

Public or normal data, on the other hand, refers to non-confidential information that is freely accessible to all users and includes things like general knowledge. Hence, rather than keeping the information in these data files private, it is crucial to make it public.

IV. IMPLEMENTATION

The proposed classification strategy is intended to handle files of various formats, such as .xml, .doc, .txt, .sql, .log, .db, .csv, .pdf,.xls and others. The strategy starts with converting the files to text format and then splitting them into smaller partitions with the HDFS input formatter. The formatter validates the input specification and generates distinct partitions, ensuring that no partition exceeds 128 MB in size. Furthermore, the reduced divides do not exceed the memory limit, and no additional memory-freeing strategies are required. After that, each partition is assigned a single Mapper for further analysis.

To classify a file as confidential or public, the security search value is analyzed across all partitions. During the creation of files that cannot be transformed into text format, such as PDFs, images, videos, and audio, their classification level is determined based on their content. After the classification process, their Enterprise Metadata Attribute (EMA) is then set as either confidential or public.In cases where a file is created without any metadata attributes, it becomes necessary to determine its classification level and assign the appropriate Enterprise Metadata Attribute (EMA) to indicate its confidentiality status.

The classification of large amounts of data involves a series of steps. Initially, the files are uploaded to HDFS for processing. Files that have an EMA value of true are considered confidential and require appropriate security measures. On the other hand, files with an EMA value of false are considered public and do not require protection. Nonetheless, files with an unknown EMA value, such as images, PDFs, audio, video, and other non-text formats, must have their classification level determined and EMA added accordingly.

The HDFS input formatter is utilized to process files that can be converted to text format, such as.txt,.doc,.xml,.csv,.xls,.sql,.log, and databases, in order to classify files that are associated with big data. This is done in order to facilitate the categorization of files. The resulting text file is then partitioned using HDFS Customized Input Format (HCIF), and each partition is analyzed independently by an HDFS Mapper. The final product is a text file.

The HDFS Reducer takes all of the classification results obtained by the Mappers and combines them into a single classification level value, where 0 indicates that the level is public and 1 indicates that it is confidential. Algorithm 1 provides an overview of the big data file classification process that has been proposed.

Algorithm 1.

Proposed Classficiation of Big Data File

Input:

Big Data File with Metadata information

Output:

File Classification as Confidential or public

Steps:

- The file is first converted into text partitions using HCIF.
    - HCIF(F) -> {P1, P2, ..., Pn}
- Each text partition is then assigned a dedicated HDFS Mapper.
    - {M1, M2, ..., Mn} = AssignMapper({P1, P2, ..., Pn})
- The HDFS Mapper is used to evaluate the content of each partition and classify it as either public or secret based on predefined policies.
    - {C1, C2, ..., Cn} = EvaluateContent({M1, M2, ..., Mn})
- Gather the classification results from all Mappers.
    - C = Aggregate({C1, C2, ..., Cn})
- Use the HDFS Reducer to combine the categorization results of all Mappers into a single result.
    - R = Combine(C)
- If the output of any Mapper is classified as confidential, the output of the Reducer will also be classified as confidential.
    - If Confidential(Ci) for any i in {1, 2, ..., n}, then Confidential(R)
- Otherwise, if all Mapper outputs are classified as public, the output of the Reducer will be classified as public as well.
    - Else if Public(Ci) for all i in {1, 2, ..., n}, then Public(R)
- If the output of the HDFS Reducer is classified as confidential, the file is marked as confidential using EMA.
    - if Confidential(R), MarkConfidential(F)
- Take care of data security during transmission.
    - EnsureDataSecurity(F)
- Otherwise, if the HDFS Reducer result is public, make the file public with EMA.
    - Else, MarkPublic(F)
- Take care of data security during transmission.
    - EnsureDataSecurity(F)

IV. RESULTS AND DISCUSSIONS

Big data can contain various types of files, including structured, semi-structured, and unstructured data, and some data may need to be made publicly accessible. To ensure secure classification of big data, decision trees are crucial. As real-time cloud applications generate more and more data, classifying this data to identify sensitive information that requires protection can be challenging. To address this, a distributed parallel decision tree approach using the Hadoop Map-Reduce architecture is employed. The decision tree has a root node and several decision nodes, and the preferred security attribute for splitting is determined using a measure function. The HDFS input formatter function utilizes this measure function to partition the massive data into multiple partitions.

To evaluate the effectiveness of the big data file classification methods, experiments were carried out on various file sizes ranging from 1 GB to 16 GB. The experiments focused on the classification time of CSV, SQL, LOG, and XLS files. The developed classification method's ability to classify files into public/secured based on different file sizes is demonstrated in **Fig 4,** which shows the corresponding classification time.

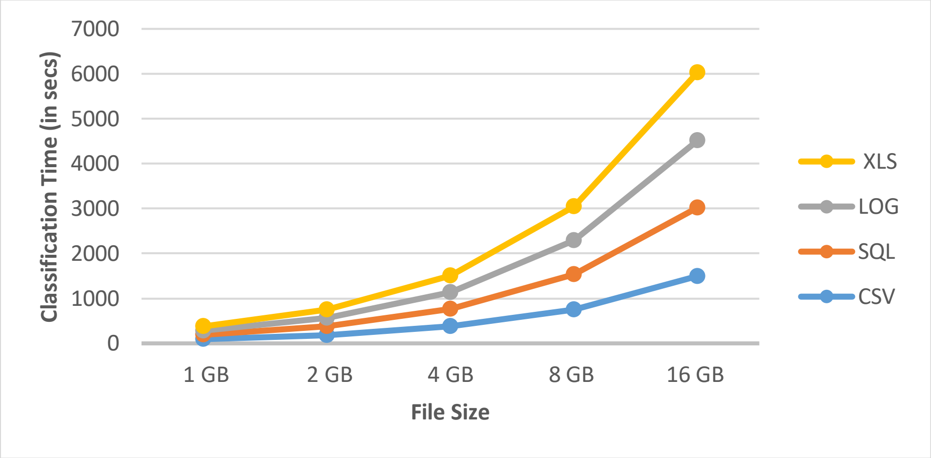

Fig 4. Comparison of Big Data File classification algorithm based on Classification Time (in seconds)

V. CONCLUSION AND FUTURE WORK

The proposed approach for ensuring data protection and security comprises of two key components: big data file classification and machine learning. The primary objective is to minimize risks while maintaining good data quality. To classify massive amounts of data efficiently, a technique was developed that leverages the HDFS MapReduce function's design to split files into fragments and distribute them among multiple mappers for efficient data verification. This classification technique identifies files requiring protection, reducing the unnecessary cost of applying data protection to public files and improving data transmission.

In order to protect big data during transmission through nodes, a security technique has been developed based on file categorization. Only files classified as confidential are secured using this technique. In future work, the extension of this security technique to include big data file classification will be explored.The efficiency of the proposed methodology is demonstrated through tabular and graphical representations that evaluate big data classification techniques based on the time taken to classify large amounts of data. The categorization method improves speed significantly by reducing repetitive encryption and decryption processes for public data.

As technology continues to evolve, future data transmission systems will require an adaptive approach that includes security measures during big data classification. Furthermore, as image, video, and audio files become more prevalent in the big data landscape, they will need to be included in the classification process. However, due to the unique properties of these file types, different strategies will need to be developed to effectively categorize them alongside traditional .txt, .csv, .log, .xls, and .sql files.

References

[1]. M. Paryasto, A. Alamsyah, B. Rahardjo et al., “Big-data security management issues,” in Information and Communication Technology (ICoICT), 2014 2nd International Conference on. IEEE, 2014, pp. 59–63.

[2]. A. K. Tiwari, H. Chaudhary, and S. Yadav, “A review on big data and its security,” in Innovations in Information, Embedded and Communication Systems (ICIIECS), 2015 International Conference on. IEEE, 2015, pp. 1–5.

[3]. Sonic, “Sonic,” last Accessed on Sept. 2018. [Online].Available:http://mirrors.sonic.net /apache/ hadoop/ common/ hadoop2.6.0/.

[4]. K. Shvachko, H. Kuang, S. Radia, and R. Chansler, “The hadoop distributed file system,” in Mass storage systems and technologies (MSST), 2010 IEEE 26th symposium on. Ieee, 2010, pp. 1–10.

[5]. J. V. Gautam, H. B. Prajapati, V. K. Dabhi, and S. Chaudhary, “A survey on job scheduling algorithms in bigdata processing,” in Electrical, Computer and Communication Technologies (ICECCT), 2015 IEEE International Conference on. IEEE, 2015, pp. 1–11.

[6]. A. Sinha and P. K. Jana, “A hybrid mapreduce-based k-means clustering using genetic algorithm for distributed datasets,” The Journal of Supercomputing, vol. 74, no. 4, pp. 1562–1579, 2018.

[7]. A. Nasridinov and Y.-H. Park, “Visual analytics for big data using r,” in Cloud and Green Computing (CGC), 2013 Third International Conference on. IEEE, 2013, pp. 564–565.

[8]. S.-H. Kim, J.-H. Eom, and T.-M. Chung, “Big data security hardening methodology using attributes relationship,” in Information Science and Applications (ICISA), 2013 International Conference on. IEEE, 2013, pp. 1–2.

[9]. A. Katal, M. Wazid, and R. Goudar, “Big data: issues, challenges, tools and good practices,” in Contemporary Computing (IC3), 2013 Sixth International Conference on. IEEE, 2013, pp. 404–409.

[10]. T. Mahmood and U. Afzal, “Security analytics: Big data analytics for cybersecurity: A review of trends, techniques and tools,” in Information assurance (ncia), 2013 2nd national conference on. IEEE, 2013, pp. 129–134.

[11]. E. Bertino and E. Ferrari, “Big data security and privacy,” in A Comprehensive Guide Through the ItalianDatabase Research Over the Last 25 Years. Springer, 2018, pp. 425–439.

[12]. V. Gadepally, B. Hancock, B. Kaiser, J. Kepner, P. Michaleas, M. Varia, and A. Yerukhimovich, “Computing on masked data to improve the security of big data,” in Technologies for Homeland Security (HST), 2015 IEEE International Symposium on. IEEE, 2015, pp. 1–6.

[13]. K. Arvind and R. Manimegalai, “Secure data classification using superior naive classifier in agent based mobile cloud computing,” Cluster Computing, vol. 20, no. 2, pp. 1535–1542, 2017.

[14]. Isaac Triguero, Daniel Peralta, JaumeBacardit, Salvador García, Francisco Herrera, MRPR: A MapReducesolution for prototype reduction in big data classification, Neurocomputing 150 (Part A) (2015) 331–345.

[15]. Simone Scardapane, Dianhui Wang, Massimo Panella, A decentralized training algorithm for echo state networks in distributed big data applications, Neural Netw. 78 (2016) 65–74.

[16]. Jemal H. Abawajy, Andrei Kelarev, Morshed Chowdhury, Large iterative multitier ensemble classifiers forsecurity of big data, IEEE Trans. Emerg. Top. Comput. 2 (3) (2014) 352–363.

[17]. Junchang Xin, Zhiqiong Wang, Luxuan Qu, Guoren Wang, Elastic extreme learning machine for big dataclassification, Neurocomputing 149 (2015) 464–471.

[18]. Alessio Bechini, Francesco Marcelloni, Armando Segatori, A MapReduce solution for associative classification of big data, Inform. Sci. 332 (2016) 33–55.

[19]. Zhenyun Deng, Xiaoshu Zhu, Debo Cheng, Ming Zong, Shichao Zhang, Efficient kNN classification algorithm for big data, Neurocomputing 195 (2016) 143–148.

[20]. Diego Marrón, Jesse Read, Albert Bifet, Nacho Navarro, Data stream classification using random featurefunctions and novel method combinations, J. Syst. Softw. 127 (2017) 195–204.

[21]. Anushree Priyadarshini, SonaliAgarwal, A map reduce based support vector machine for big data classification, Int. J. Database Theory Appl. 8 (5) (2015) 77–98.

[22]. SeyedaliMirjalili, Seyed Mohammad Mirjalili, Andrew Lewis, Grey wolf optimizer, Adv. Eng. Softw. 69(2014) 46–61.

[23]. A. Al-Shomrani, F. Fathy, and K. Jambi, “Policy enforcement for big data security,” in Anti-Cyber Crimes(ICACC), 2017 2nd International Conference on. IEEE, 2017, pp. 70–74.

[24]. S.-H. Kim, N.-U. Kim, and T.-M. Chung, “Attribute relationship evaluation methodology for big data security,” in IT Convergence and Security (ICITCS), 2013 International Conference on. IEEE, 2013, pp. 1–

[25]. B. Cruz, “Vulnerability, exposure, threat and risk terms,” last Accessed on Sept. 2018. [Online]. Available:http://belencruz.com/en/2013/04/ vulnerability-exposure-threat-and-risk-terms/.

[26]. T. M. Corporation, “Common vulnerabilities and exposures,” last Accessed on August 2018. [Online]. Available: https://cve.mitre.org/cve/.

[27]. L. Hayden, IT security metrics: A practical framework for measuring security &amp; protecting data. McGraw-Hill Education Group, 2010.