Big Data Processing Based on Machine Learning for Multi-user Environments

Kamel H. Rahouma and Farag M. Afify

Abstract Many sources of data yield non-structured data like the Internet of things (IoT), geospatial data, E-commerce, social media, and scientific research that is not appropriate in to traditional, structured warehouses. Nowadays, sophisticated analytical techniques allow companies to obtain perspicacity from data with earlier unachievable levels of accuracy and speed. Real-time analytics for big data is the capability to achieve the most suitable decisions and get significant actions at the best time. First, we present a survey of processing the big data (BD) in real time (RT) and focus on its challenges. Then, we propose an algorithm to handle BD by integration with machine learning operations in multi-user environment optimization operations, reduce maintenance costs and better speed of fault detector and provide common operations necessary to process unstructured information. There are important conditions that have been taken into a concern to guarantee the quality of services (QoS) and transmission velocity and ensure the system's physical time synchronization and the correctness of the data processing.

Keywords Big data · Multi-user environment · Real-time processing · Machine learning

1 Introduction

The developments in the application of scientific knowledge are for practical purposes, especially in an industry in the last few years driven to the accumulation of huge quantities of information. BD brings recognition from business, governments' operations, research, etc., because BD tools let users treat big amounts of information to obtain the best judgment through the analytics of information. Machine learning (ML) is a vital element in the analytic of information as it prepares the machines to obtain knowledge by using trained information, then it makes efficient decisions by using experiments that decreases the time of data processing.

K. H. Rahouma (✉) · F. M. Afify
Electrical Engineering Department, Faculty of Engineering, Minia University, Minia, Egypt
e-mail: kamel_rahouma@yahoo.com

D. K. Sharma et al. (eds.), *Micro-Electronics and Telecommunication Engineering*, Lecture Notes in Networks and Systems 106,
https://doi.org/10.1007/978-981-15-2329-8_68

To obtain certain decisions, ML tools were employed, and it can increase the performance of other components of knowledge innovation.

BD [1] has grown within the presence for a few years. BD is related to traditional database systems, though the distinction is that it exceeds in volume and processing.

BD is very huge, transfers so quick including unorganized and organized information, while traditional data contain the only structured information [2].

The digitization of all operations now produces different kinds of huge and RT data across a wide range of processes. Several applications produce BD, for example, social media, scientific experiments, cloud applications, E-government services, monitoring programs, and information warehouses. Data expand quickly since operations generate continuously rising amounts of unorganized and structured information [3]. The impact on information storage, processing, and transfer is necessary to re-evaluate the methods and clarifications to great result the user wants. In that context, processing types and algorithms have a great role. A great quality of answers for special applications and programs exist, so an accurate and systematic analysis of present solutions for processing standards, algorithms, and techniques applied in BD storage and processing environments has great importance. Nowadays, sophisticated analytical techniques allow companies to obtain perspicacity from data with earlier unachievable levels of accuracy and speed. Real-time analytics for BD is the capability to achieve the most beneficial judgments and get significant actions at the best time [4].

In this paper, we present an overview of processing the BD in RT and focus on its challenges. We suggest an algorithm to handle BD in a multi-user framework, and we provide common operations necessary to process unstructured data. The paper also provides an algorithm to schedule multiple functions of users simultaneously in a distributed environment. The rest of the paper introduces a background of the big data in Sect. 2. In Sect. 3, we give a literature review, and in Sect. 4 we explain the methodology of this paper. In Sect. 5, we depict the outcome of our work and compare our results and the previous work results. In Sect. 6, we introduce some of the conclusions, and a list of the used references is given at the end of the paper.

2 A Background About BD

2.1 What Is BD?

Among all the descriptions given for "BD," commonly it means information that is very huge, very fast, or very difficult for existing tools to process. "Very huge" shows that companies frequently and necessarily agree to deal with petabyte-scale quantities of information that become of sensors, transaction histories, clicks streams, and outside. "Very fast" shows that not only is information huge, but also it is to be processed immediately. For example, it is to make cheating detection at a location of sale or to determine which ad to display to a person on a webpage. "Very difficult" is a collection of information that does not match cleverly into a current processing tool or that needs some kind of analysis that current tools cannot offer [5].

2.2 Characteristics of BD

There are three Vs which may shortly describe the characteristics of the BD. These are the variety, velocity, and volume. The volume means the number of datasets and storage. The variety means the information types. The velocity means the rapidity of incoming information. Growth of research and discussions concern with BD, and the three Vs were expanded to five Vs as shown in Fig. 1 These are the veracity, variety, volume, velocity, and value [6].

Today's BD plays an important function for a lot of fields like a business, online purchasing, banking, astronomy, health care, and finance. BD gives great advantages to business activities. BD is produced by a huge quantity of data. These data extend to rise every day because information arrives permanently of various operations.

BD has a huge quantity of unwanted data in both unorganized and organized structures. In unorganized, the information is stored in undefined and unsystematic ways, while organized information is saved into well-defined structures [7].

Facebook, Wikipedia, and Google produced unorganized information, while E-commerce operations produced organized information. Since the appearance of organized and unorganized information, some difficulties appear in BD such as data collecting, data transfer, sharing, storage, privacy, analysis, search, handling of information, fault tolerance, and visualization. It is impossible to handle these difficulties in regular ways. Regular information administration mechanisms were unable to process, analyze, and schedule jobs in BD. Therefore, BD uses different tools to handle these challenges. This paper aims to study these challenges [8].

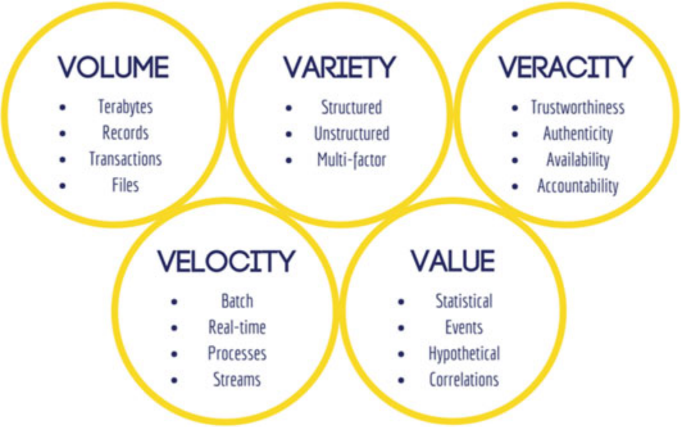

Fig. 1 Five Vs of big data

2.3 BD Software

2.3.1 Hadoop (HD)

The appropriate software for managing BD difficulties is HD. HD is software that allows distributed handling of huge information across clusters of machines applying simple principles of programming [9]. It contains two parts. The first part is used for storage, and it is named Hadoop Distributed File System (HDFS). The second part is employed for processing, and it is named MapReduce (MR). The HDFS has a master structure and slave structure. The master is a process named (NameNode) which controls the operations on files and manages the global namespace. The slave is a process named (DataNode), and it saves the data in the structure of data blocks and operations as instructed by the NameNode. The NameNode handles the information replication and arrangement for reliability, fault tolerance, and performance. The NameNode divides and saves files into 64 MB data blocks over the DataNodes. Typically, there are three duplicates of all data blocks which are saved in the HDFS. The failure detection is implemented in the framework of regular duplicates of DataNodes to NameNode. If there is no heartbeat from a DataNode for a long time, it is marked as lost not employed for new processes and if required, additional duplicates of its data are implemented [10].

2.3.2 MapReduce (MR)

MR is an information handling standard for dealing with various challenges of computing. The MR concept is stimulated through the map, and it decreases functions, which are usually used in functional languages. The MR allows administrators to simply display their process like a map and decrease operations [11].

2.4 BD Processing Framework Challenges

Increasing the number of researches in the BD field does not express that we understand all BD processes, so we do not have a common definition of BD and we have a lot of differences about tools and applications [12]. Moreover, there is a big problem that researchers are currently interested, this problem is process the information in RT and the process by which the data we need to make a quick decision is processed. like monitoring and protecting systems, Electrical grid and Smart Cities [13–15]:

1. Manage energy of all sectors.
2. Solve any faults problems.
3. Save energy as much as possible.
4. Distribute power reasonably.

To execute these Jobs, fault analysis and error status must perform during a low period; oppositely that make more failures in the electric network. RTBD has important requirements in analysis, acquisition, security, data management, and benchmarking. These challenges are briefly described in the following [16–18]:

1. The RT processing speed.
2. The RT systems stability.
3. The large-scale applications.
4. Data collection.
5. Data analytics.
6. Data security.
7. Usability issue of data management.
8. Test benchmark of performance.

2.5 Technical Challenges

Added to the last challenges, there are some technical challenges which may be faced by BD systems. From these challenges [19, 20]:

1. Fault tolerance

New incoming technologies like cloud computing and BD it is regularly that whenever the failure happens the damage is done should be within acceptable threshold rather than beginning the whole job from scratch. Fault-tolerant computing is greatly complicated, containing complex algorithms. Hence, the main responsibility is to reduce the possibility of error to the minimal level.

2. Quality of data

Storage and collection of a huge amount of information are more costly. More information is applied for predictive analysis and decision making in business will give to best results. BD focuses on the quality of information storage rather than having very great irreverent information so that excellent result and conclusion can be formed.

3. Heterogeneous data

In BD, unstructured information describes almost every sort of data being generated by fax transfer, social media interaction, and handles various kinds of document and more. Transforming unorganized information into organized information one is also not possible. Structured information is organized but unstructured information is raw and unorganized. In addition, two important challenges are visualization and hidden BD.

3 A Literature Review

Data is one of the most important things in all parts of our lives, and every day large quantities are produced. Many international reports indicate that data quantities are multiplying daily and very quickly. Despite the huge flood of data, they are only used to a limited extent where they are managed, processed, and analyzed for their use [21]. These huge quantities need to be dealt quickly, and some of them require real-time processing, for example, detect fraud and data for security institutions in general, and there are companies benefited from these operations, especially fraud operations such as PayPal. As a result of the great need and difficulties to deal with these huge amounts of data appeared, many tools to deal with HD and MR; the major companies are working to take advantage of these quantities like Google and Yahoo. In the past few years, the research has focused on big data operations, and a large number of tools have been proposed, for example, Storm [22] and Spark [23]; these tools are able to face many operations challenges dealt with on BD. These tools help companies in data processing and obtaining the desired results, and it has been used in many large companies such as Netflix which give them the best recommendations to deal with customers and address errors.

Feldman et al suggested suitable methods for handling big data and provided a suitable solution for the large storage areas that are used daily to store this data. Data compression has become an important element that has proven to be effective when used by Feldman and all data still without a loss. Faced with the difficulties of handling BD was behind the suggestion that was introduced by Jiang et al. A tool capable of handling and retrieving big data, increasing processing speed, analyzing, and assisting in decision making was introduced. Cuzzocrea et al proposed a tool capable of handling graphs and data for BigWeb Data. This tool has proven to be effective and has provided a solution to most of the problems facing companies collecting data from the web [24–26].

4 The Methodology

Our methodology depends on using ML; it is a combination of a set of sciences which able us to get benefit on existing data and learn and act on previous experiences. At the moment, ML is used in all aspects of life such as health, industry, and trade. ML technologies enable the handling of layers, devices, and data that were impossible to access, make decisions, and avoid previous errors. Data processing is usually required before it can be used to remove unnecessary data and incomplete data through which systems can learn to avoid past errors, increase productivity, and develop products [27, 28]. ML tools have been able to gain experience by learning and benefit from the experiences built by the systems and the way we can take advantage of the data. If we say that the big data helps us to store large amounts of data, through ML and

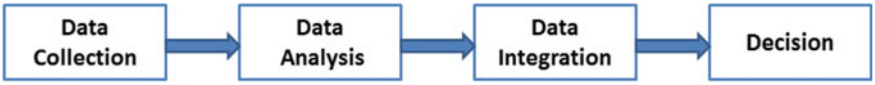

Fig. 2 Real-time data processing stages

integration with BD as shown in Fig. 4, we can get useful information that is easy to use in all areas [29].

4.1 The Proposed Algorithm

We proposed an algorithm with a high-level programming interface which achieves integration between BD and ML and provides common operations necessary to process unstructured data. It also supports an algorithm to schedule multiple functions of users simultaneously in a distributed environment. It has also been taken into account robust computing capability; a large RT information processing necessitates a powerful timing, and this means the system must respond to any request in the shortest possible time. So in the beginning, the RT data processing system must have powerful data computing capability. The traditional way of processing large data is depending on the robust computing abilities of a cloud to obtain the desired goal, while at the RT, it necessitates depending on the strength to rapidly exchange information between devices.

The RT data processing framework is split into four stages: collection, analysis, integration of data, and resolution as shown in Fig. 2.

Data collection is responsible for information collecting plus storage and includes information cleaning and information preparation for analysis.

Data analysis is the core of the large real-time data processing system and the critical stage to determine system performance. This phase is mainly responsible for modeling data structures, clearing data, processing, and making information ready for integration layer.

Data integration: This phase plays a related role in these processes. At this stage, the combination is between more information processing algorithms and providing support for other layers.

Decision making: In this layer, decisions are made based on the data coming from other layers, from which the final objective of the information analysis process is produced.

This algorithm will help to find the similarities between the documents to make informed decisions about clustering. Many distance metrics can be used to find similarities between documents.

The framework described generically in nature works with any set of documents that make up large data. The frame takes large data as inputs and produces clusters. The processes involved in the framework include keyword selection, creation of feature space, calculation of similarity, and aggregation.

The algorithm is responsible for terminating specific tasks to improve the productivity of large data processing. The distributed environment takes several useful functions and arranges them correctly in a way that is handled optimally. The concept of waiting time is used to ensure that jobs are given their role, and large data processing is performed efficiently.

4.1.1 Algorithm

```
Itialize files = Get list of files in storage(Partition)
Itialize filesData = Get last running data result(DB)
Itialize newFileData<File, HashCode>.
foreach file in files

  newHashCode = generateFileHashCode(file)
  oldHashCode = getSavedHashCode(file, filesData)
  AddRecode(newFileData, file, newHashCode)
  if oldHashCode = nothing
    Report("New Added File");
  elseIf oldHashCode = newHashCode
    Report("no Change on File")
  Else
    Report ("File Updated")
  EndIf

next file
forEach file in filesData

  If files not contains file
    Report("File Deleted")
  End if

Next file
newFileData Overwrite fileData

  Save(fileData)
```

4.1.2 The Scheduling

We suggest Responsive Job Scheduler that depends on the position of reference. The suggested tool intended for a group contains machines or nodes (N) from 1 to *n*, and these machines contain what it needs to work efficiently. The group contains a multi slave node (SN) and one master node (MN). MN contains operations list and handling these operations according to SN specification.

The workflow of operations list is as follows:

1. The new job comes to operations list (OL).
2. Task is split into two tasks (Map and Reduce) and map split to local non-local job.
3. When a new slave task is added, then it checks for free slot count.
4. If any job arrives at SN, it sends a signal to MN which manages operations list for SN.
5. MN inspects the arriving job to determine the appropriate time for implementation and waiting for finishing tasks or arriving jobs.

$$\text{Data locality} = \text{no .of local map/task Total map task}$$

The fairness of a job

$$\text{fairshare} = \text{job weight}/\varSigma \text{ job weight} * \text{capacity of taktracker}$$

Average acknowledgment time

$$T_{\text{avg}}{=}Rl * T_{\text{avg}}\ 1 + (1 - Rl)\ T_{\text{avg}}^{n\text{th}}$$

Performance =

$$\text{Map task} = \text{no.of pending map tasks/no.of currently running map task}$$

$$\text{Reduce task} = \text{no.of pending reduce tasks/no.of currently running reduce task}$$

The new framework is a combination of algorithms to execute the RTBD analytics and to improve the precision of information which the supervisor needs it. The proposed framework, based on these algorithms, enables the supervisor to work on BD and face constraints and discover useful data and decisions.

5 The Results

RT forecasting of complex systems is important because it is easy to plan maintenance, minimize problems caused by sudden downtime, and reduce the value of spare parts consumed in maintenance. This proposal uses BD integration and machine learning processes.

In this case, the focus was on the data which come from the monitoring of the status and power consumption of motors and sensors which is a vital part of the operation in baggage handling systems and describes the experiments carried out to estimate the efficiency of the suggested process. Figure 3 presents the groups' status and power consumption motors and sensors.

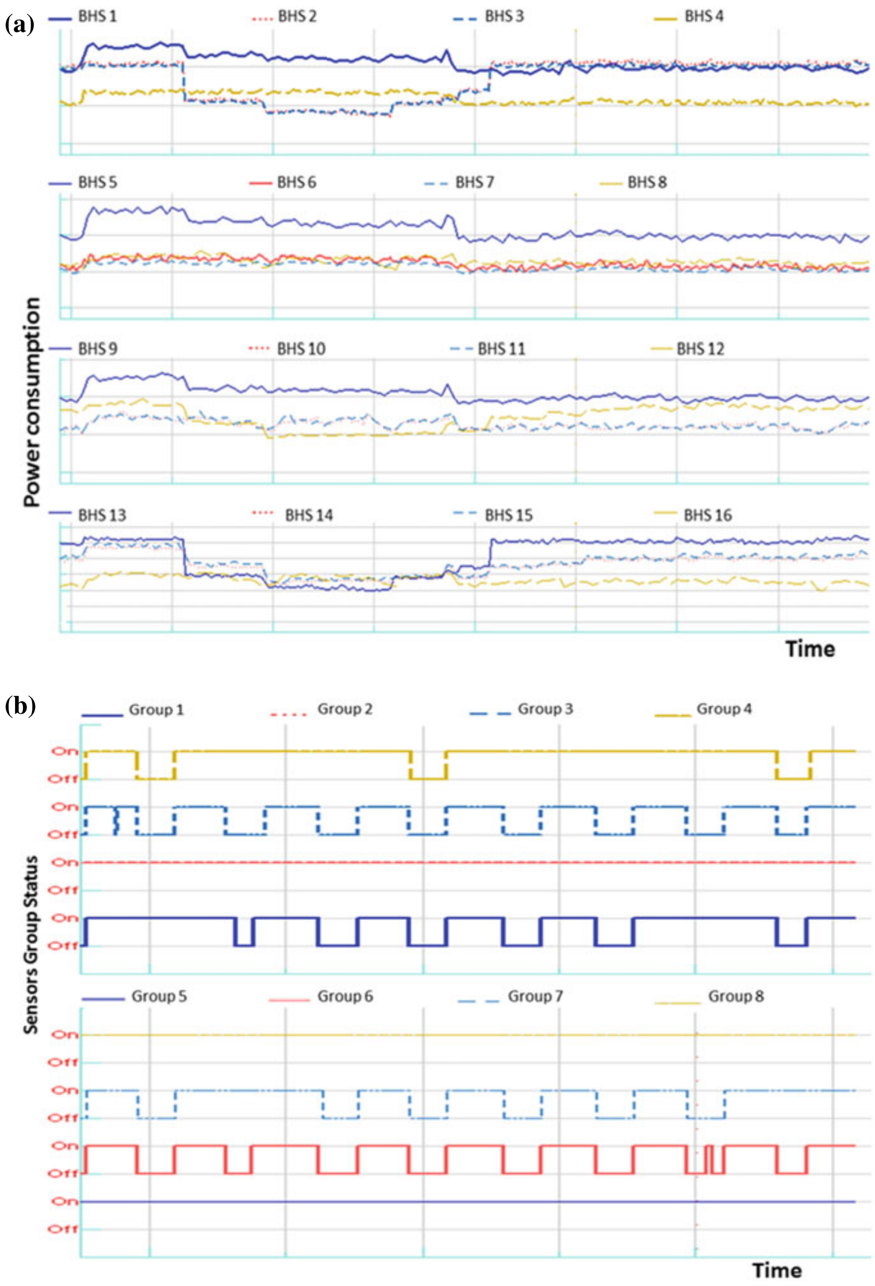

Fig. 3 **a** Phase of RT information processing and detect faults in BHS. **b** Phase of RT information processing and detect faults in sensors group

The tool was able to perform real-time processes on gram amounts of data, exploiting the benefits of working on unstructured data in fault detection and decision making to decrease solving problem time.

The experimental results showed a rapid increase in the amount of information that has been processed, in addition to the increase in the amount of data that did not affect the implementation times.

Using this tool provides also a fault ranking estimation, and this is particularly useful for this application on BHS where, almost always, an unbalance fault shows up as well, also if another error is the root cause of the anomaly. Examining the assignment features is then a true benefit of the tool. The balance between the amount of information being worked on and the number of devices must be considered to achieve the best performance. The present technique is found to be more efficient than the existing ones in the literature. Some of the reasons for that are as follows:

1. Most existing techniques have the inherent problems of analyzing unstructured data in real time, and the operations of these techniques depend on collected data for systems especially industrial systems. In our proposal, the algorithm capable of executing operations on huge quantities of unorganized information in real time and increasing the efficiency of fault detection reduces the extra cost if the system has failed down during executing important jobs and improvements of the efficiency of BD operations.
2. The tool reduces maintenance time according to the status, saving of running tasks time when detecting the failure and faults prediction if any data come from sensor contrary to the normal situation.
3. The tool enables data-based decision making because it creates a value of information previously neglected by handling the problems quickly and gives a better understanding and sufficient detail to these troubles.

6 Conclusions

This paper provided an overview of processing BD in RT and focused on its challenges. An algorithm is suggested to handle big data by integration with machine learning operations in a multi-user environment optimization. This reduced the maintenance costs and resulted in a better speed of fault detector and provided common operations necessary to process unstructured data. The network transmission speed and the quality of service (QoS) factors have been taken into account. Thus, we make sure that the system works efficiently and make sure that the data processing process is completed efficiently.

References

1. Kshetri N (2014) Big data s impact on privacy, security and consumer welfare. Telecommun Policy 38(11):1134–1145
2. Kitchin R (2014) The real-time city? Big data and smart urbanism. Geo J 1–14
3. Chang RM, Kauffman RJ, Kwon YO (2014) Understanding the paradigm shift to computational social science in the presence of big data. Decis Support Syst 63:67–80
4. Chen J, Chen Y, Du X et al (2013) Big data challenge: a data management perspective. Front Comput Sci 7(2):157–164
5. Zhao JM, Wang WS, Liu X et al (2014) Big data benchmark big DS. Advancing Big Data Benchmarks. Springer International Publishing, pp 49–57
6. Wang CC, Chen CL, Hou ZY et al (2015) A 60 V tolerance transceiver with ESD protection for FlexRay-based communication systems. IEEE Trans Circ Syst I: Regul Pap 62(3):752–760
7. Hwang K, Chen M (2017) Big data analytics for cloud/IoT and cognitive learning. Wiley, UK
8. Hwang K, Chen M, Wu J (2016) Mobile big data management and innovative applications (editorial). IEEE Trans Serv Comput 9(5):784–785
9. Bende S, Shedge R (2016) Dealing with small files problem in hadoop distributed file system. Procedia Comput Sci 79:1001–1012
10. Hadoop Distributed File System [online]. http://hadoop.apache.org/hdfs
11. Cheng D, Zhou X, Lama P, Wu J, Jiang C (2017) Cross-platform resource scheduling for spark and MapReduce on YARN. IEEE Trans Comput 66:1341
12. Wang B, Jiang J, Wu Y, Yang G, Li K (2016) Accelerating MapReduce on commodity clusters: an SSD-empowered approach. In: IEEE transactions on big data, IEEE, 2016
13. Y. Liu, M. Qiu, C. Liu, et al., Big data challenges in ocean observation: a survey, Personal Ubiquitous Comput. 2017
14. Yildiz O, Ibrahim S, Antoniu G (2017) Enabling fast failure recovery in shared Hadoopclusters: towards failure-aware scheduling. Future Gener Comput Syst 74:208–219
15. Mavridis I, Karatza H (2017) Performance evaluation of cloud-based log file analysis with Apache Hadoop and Apache Spark. J Syst Softw 125:133–151
16. Xu H, Lau WC (2017) Optimization for speculative execution in big data processing clusters. IEEE Trans Parallel Distrib Syst 28(2):530–545
17. Gani A, Siddiqa A, Shamshirband S, Hanum F (2016) A survey on indexing techniques for big data: taxonomy and performance evaluation. Knowl Inf Syst 46(2):241–284
18. Gil D, Song I-Y (2016) Modeling and management of big data: challenges and opportunities. Future Gener Comput Syst 63:96–99
19. Xu H, Lau WC (2017) Optimization for speculative execution in big data processing clusters. IEEE Trans Parallel Distrib Syst 28(2):530–545
20. Sivarajah Uthayasankar (2017) Muhammad Mustafa Kamal, Zahir Irani, Vishanth Weerakkody, Critical analysis of Big Data challenges and analytical methods. J Bus Res 70:263–286
21. Zhou L, Pan S, Wang J, Vasilakos AV (2017) Machine learning on big data: opportunities and challenges. Neurocomputing 237(10 May):350–361
22. https://storm.apache.org/. Accessed July 2019
23. https://spark.apache.org/. Accessed July 2019
24. Yokota R, Wu W (eds) (2018) Supercomputing frontiers. In: 4th Asian conference, SCFA 2018 Singapore, 26–29 March 2018
25. Feldman D, Schmidt M, Sohler C (2013) Turning big data into tiny data: constant-size coresets for k-means, PCA and projective clustering. In: SODA, 2013
26. Jiang F, Leung CK (2015) A data analytic algorithm for managing, querying, and processing uncertain big data in cloud environments. Algorithms 8:1175–1194

27. Cuzzocrea A, Cosulschi M, De Virgilio R (2016) An effective and efficient MapReduce algorithm for computing BFS-based traversals of large-scale RDF graphs. Algorithms
28. Jordan MI, Mitchell TM (2015) Machine learning: trends, perspectives, and prospects. Science 349:255–260
29. Zhou L, Pan S, Wang J, Vasilakos AV (2017) Machine learning on big data: opportunities and challenges. Neurocomputing 237(10):350–361