Artificial Intelligence Review (2020) 53:989–1037
https://doi.org/10.1007/s10462-019-09685-9

The state of the art and taxonomy of big data analytics: view from new big data framework

Azlinah Mohamed[1] · Maryam Khanian Najafabadi[2] · Yap Bee Wah[1] · Ezzatul Akmal Kamaru Zaman[1] · Ruhaila Maskat[1]

Published online: 1 February 2019

Abstract
Big data has become a significant research area due to the birth of enormous data generated from various sources like social media, internet of things and multimedia applications. Big data has played critical role in many decision makings and forecasting domains such as recommendation systems, business analysis, healthcare, web display advertising, clinicians, transportation, fraud detection and tourism marketing. The rapid development of various big data tools such as Hadoop, Storm, Spark, Flink, Kafka and Pig in research and industrial communities has allowed the huge number of data to be distributed, communicated and processed. Big data applications use big data analytics techniques to efficiently analyze large amounts of data. However, choosing the suitable big data tools based on batch and stream data processing and analytics techniques for development a big data system are difficult due to the challenges in processing and applying big data. Practitioners and researchers who are developing big data systems have inadequate information about the current technology and requirement concerning the big data platform. Hence, the strengths and weaknesses of big data technologies and effective solutions for Big Data challenges are needed to be discussed. Hence, due to that, this paper presents a review of the literature that analyzes the use of big data tools and big data analytics techniques in areas like health and medical care, social networking and internet, government and public sector, natural resource management, economic and business sector. The goals of this paper are to (1) understand the trend of big data-related research and current frames of big data technologies; (2) identify trends in the use or research of big data tools based on batch and stream processing and big data analytics techniques; (3) assist and provide new researchers and practitioners to place new research activity in this domain appropriately. The findings of this study will provide insights and knowledge on the existing big data platforms and their application domains, the advantages and disadvantages of big data tools, big data analytics techniques and their use, and new research opportunities in future development of big data systems.

Keywords Parallel and distributed computing · Big data tools · Big data analytics techniques · Domain area

✉ Maryam Khanian Najafabadi
maryam.najafabadi@newinti.edu.my

Extended author information available on the last page of the article

1 Introduction

Big data is becoming more significant in a large number of research areas like social networks, semantic Web, data mining, information fusion, computational intelligence and machine learning. In order to deal with large amount of data to be processed by a single computer, it is necessary to employ parallel and distributed computing (Prajapati et al. 2017; Plimpton and Shead 2014). The rapid development of various big data tools such as Hadoop, Storm, Spark, Flink, Mesos and etc. in research and industrial communities has allowed the huge number of data to be distributed, communicated and processed. Apache Hadoop and, more recently Spark have been very well known frameworks for massive amounts of data processing based on the MapReduce paradigm that allows for efficient utilization of data mining and machine learning techniques in different domains (Arias et al. 2017; Kousiouris et al. 2018). Data mining techniques running on MapReduce paradigm utilizes a propitious environment to successfully deliver results quickly while maintaining a high throughput. In fact, a growing open source projects, called SparkMLib and Mahout have been designed to supply an implementation of distributed and scalable data mining algorithms such as naïve-bayes classifier, nearest neighbor and k-means clustering. The combination of big data tools and big data analytics techniques especially machine learning and data mining algorithms has generated interesting and new challenges in areas such as public services (education, health care, and transportation), social media and social networks. These new challenges tackle problems in data storage, data processing, tracking data, analyzing user behaviours and data utilization for pattern mining, and data visualizing (Wang and Belhassena 2017; Spivak et al. 2018).

There are literature reviews on big data that have been published. However, these literature reviews focus on either a specific domain of big data application or big data analytics techniques; none of these articles concentrate on the comprehensive analysis of the application of big data technologies. This paper presents a survey of recent technical works in dealing with requirements and technologies considered for big data platforms. In order to gain insight on big data platforms, we have reviewed and classified articles on big data that have been published in academic journals. This work of literature review is hoped to support researchers and practitioners in selecting and adopting big data platforms according to their technological needs and specific application requirements. It provides not only a global view of main Big Data technologies but also popular Big Data tools that are of three categories, which are batch processing, stream processing and hybrid processing tools (Prajapati et al. 2017). Under batch processing, a set of data or a batch of information is collected over time, and then fed into an analytics system for processing. Under the big data stream computing, processing is done in real time as data is fed into analytics tools piece-by-piece. Batch processing is a model of storing then computing whose milestone is MapReduce framework developed by Google in 2003 while stream processing is a model of straight through computing whose milestone is the S4 proposed by Yahoo! in 2010. However, big data batch computing cannot support big data stream computing. The characteristics of streaming processing are: (1) High volumes refer to the volumes of arriving of continuous data that is beyond the capabilities of individual machines; (2) Low latency occurs when the input data streams produce output streams during multi-staged computing. The third generation of big data tool is called hybrid processing and it can be advantageous due to its capability for both batch processing and stream processing (Sahal et al. 2017; Rathore et al. 2016).

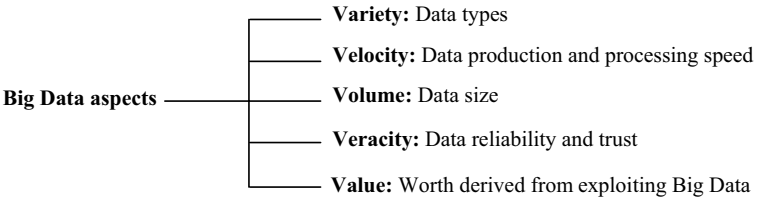

Fig. 1 Big data aspects

This paper makes several significant contributions. First, it provides a comprehensive review on the considered requirement and technology for big data management based on the works taken from the Science Direct database and this included 93 most searched articles. Second, all 93 related articles on big data have been reviewed, analyzed, and then classified with respect to the application of big data in real life or domain areas, sources of data generation, big data tools for data processing and storage and big data analytics techniques. Most importantly, this paper proposes a classification scheme to examine and classify the studies conducted in big data with respect to distinct processing paradigms including batch processing, stream processing and hybrid processing. Third, different domain areas in big data and their concept of big data tools have been explained in this research work. Finally, the research implications and new research opportunities have been stated to provide new research directions and opportunities in the big data area. Hence, in doing all these, this paper is structured into several sections. Section 2 provides the definitions and main concepts of big data. Section 3 describes the research methodology, the search strategy, the selection process of papers based on evaluation criteria and data extraction strategy. Section 4 describes the results of the review conducted. Section 5 discusses the distribution of research papers based on classification method proposed. Section 6 concludes the report by summarizing the results and highlighting some ideas on future work.

2 Big data definition and challenges

There are five major aspects of big data: (Refer to Fig. 1.):

(1) Variety refers to the different form of data types and sources such as textual information about active events; traffic dataset about vehicular traffic on roads and scheduled events (e.g., music events, sporting events). (2) Velocity shows the speed of data in and out. It refers to the dynamic feature of data, the frequency of data generation, and the necessity of generating results in real-time. (3) of Big Data volume focuses on the size of data set that reaches the level of megabytes or gigabytes, terabytes or even petabytes. (4) Veracity refers to the extent of the data can be trusted, given the reliability of its source. For instance, when receiving data from sensors, some devices may be compromised. (5) Value corresponds to potential insights that an organization can derive from processing big data referring either to the big potential value or the extremely low density of value (Kranjc et al. 2017; Liang et al. 2016; Zhang et al. 2016).

Many computational platforms and tools have been proposed for handling big data. The most known platform designed for large scale processing on clusters of commodity hardware is Apache Hadoop to implement MapReduce programming model. MapReduce programming model processes large data sets via developing parallel and distributed algorithms (Prajapati et al. 2017; Kumar and Rath 2015). In general, Hadoop stores data sets

across distributed clusters in order to run a distributed processing scheme in each cluster. Hadoop and MapReduce follow a batch processing model in which computations start and end within a given time frame. In spite of successful of Hadoop in retrieving analytics and building decision making mechanisms when data are 'static, it has problems when applications require real-time management of data streams. For example, a large company wants to identify patterns of the buyers' behavior. The company uses Hadoop to analyze the historical data of buyers' behavior to retrieve the discussed patterns in vast amounts of data and take specific decisions for the future strategy. In this case, real-time data stream processing cannot be the appropriate technique. Now consider on a power plant, where security issues are very important. Sensors for natural phenomena (e.g., people movement, temperature) create large amounts of data that should be supported in real-time. In this case, alarms should be created when specific criteria are met and decisions should be taken based on the response to security violations. In lieu with this, big data analytics require real-time data stream processing for handling the speed of data arrival, data management and data storage. Hence, Apache SPARK emerges to overcome problems of batch processing by extending Hadoop with new workloads like interactive queries, streaming and learning algorithms. However, the processing in a streaming computing scenario is open-ended and the program is able to process documents forever while support high levels of data throughput and a low level of response latency (Bei et al. 2018; Wang et al. 2017a, b).

The volatile growth of data based on rapid development of Cloud Computing, Internet of Things, and Internet has led to several challenges. These challenges are like the manner or the way of capturing massive volume of data, the manner in which the massive data can be stored in a limited memory space assigned and the manner that the data can be processed, transferred and analyzed in the absence of intelligent algorithm. Furthermore, standard reduction techniques cannot support large-scale data since their runtime becomes impractical. Several other approaches have been tackled in enabling data reduction techniques, which can solve this problem. In this research work, we will present how researchers address these challenges by considering the numbers of publications in big data. We will extract the data from the articles to answer challenges mentioned. The results of this survey will provide guidelines for future research on big data frameworks.

3 Research methodology

Big data has become an important research field in information sciences, commercial activities, policy, public administration and decision makers in governments and enterprises. It also leads to new research paradigms and future businesses that focus on big data. Hence, the objective of this research is to understand the requirements and technologies considered for big data platform by examining the published articles, and to provide researchers and practitioners with insights and future directions of big data. Hence, we will verify the distribution of articles on big data, and classify the articles by domain areas of big data applications, data sources, data processing and storage of data and techniques used for data analytics. However, we consulted the Journal of Citation Reports (JCR) for the top journals in Science Direct, as a cross-reference to ensure that we had captured results from the top journals. There were 18 journals listed in JCR and we searched the distribution of articles published by these journals. Our main criterion for conducting the literature search was the use of term "big data platform" as the core argument developed or the core technology analyzed in the paper, typically evidenced by its emphasis in the title, abstract and/

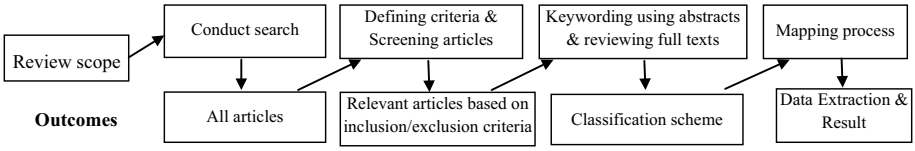

Fig. 2 Research methodology

or keywords. Our search identified 93 articles from journals that had the highest quality in big data field. The full text of each article was reviewed to eliminate those that were not in accordance with the focus of this survey. A set of inclusion/exclusions criteria based on guideline proposed by Najafabadi and Mahrin (2016) and Najafabadi et al. (2017a, b) to limit our collection of the articles were:

1. To achieve optimum number of papers that were being cited in big data area, only articles that were related to the big data community research works were searched for. Thus, master and doctoral dissertations, poster presentation, conference papers, non-English papers, textbooks, and literatures that cannot answer to objective of this research were not included in this survey.
2. Only those articles which clearly described the big data tools including Storm, Spark, Hadoop, Flume, Flink, Pig, S4, Kafka were selected to construct the frame of big data technologies.
3. Each article was carefully reviewed and separately classified according to the four categories of our big data framework which were domain areas, sources of data generation, big data tools for data processing and storage of data. The details of the big data analytics techniques will be presented in the next section (Sect. 4). This search serves as a comprehensive base to create an understanding of big data research in the construction of frame of big data technologies for different domains such as smart city, network security, healthcare applications, transportation management, government and public sector, social networking, energy consumption, education and research, finance and fraud detection.
4. Based on our investigation, we derived the following records of each article: (the articles that are not able to explicitly answer the four research questions will be excluded.)
   - What are the domain areas and data sources in big data frameworks proposed in each articles?
   - What are the datasets used for evaluating the proposed big data frameworks?
   - What are the existing big data tools for data processing and storage?
   - What are the most current big data techniques used in big data frameworks?

4 Classification method based on various aspects of big data life

Figure 2 depicts the research methodology used in conducting this survey.

Our classification framework consists of domain areas or application fields of big data in real life, data sources, big data tools for data processing and techniques used for data analytics. In this research, we classify the articles that were reviewed into five categories of domain areas and three main categories of big data tools including batch processing, stream processing and hybrid processing and techniques used for big data

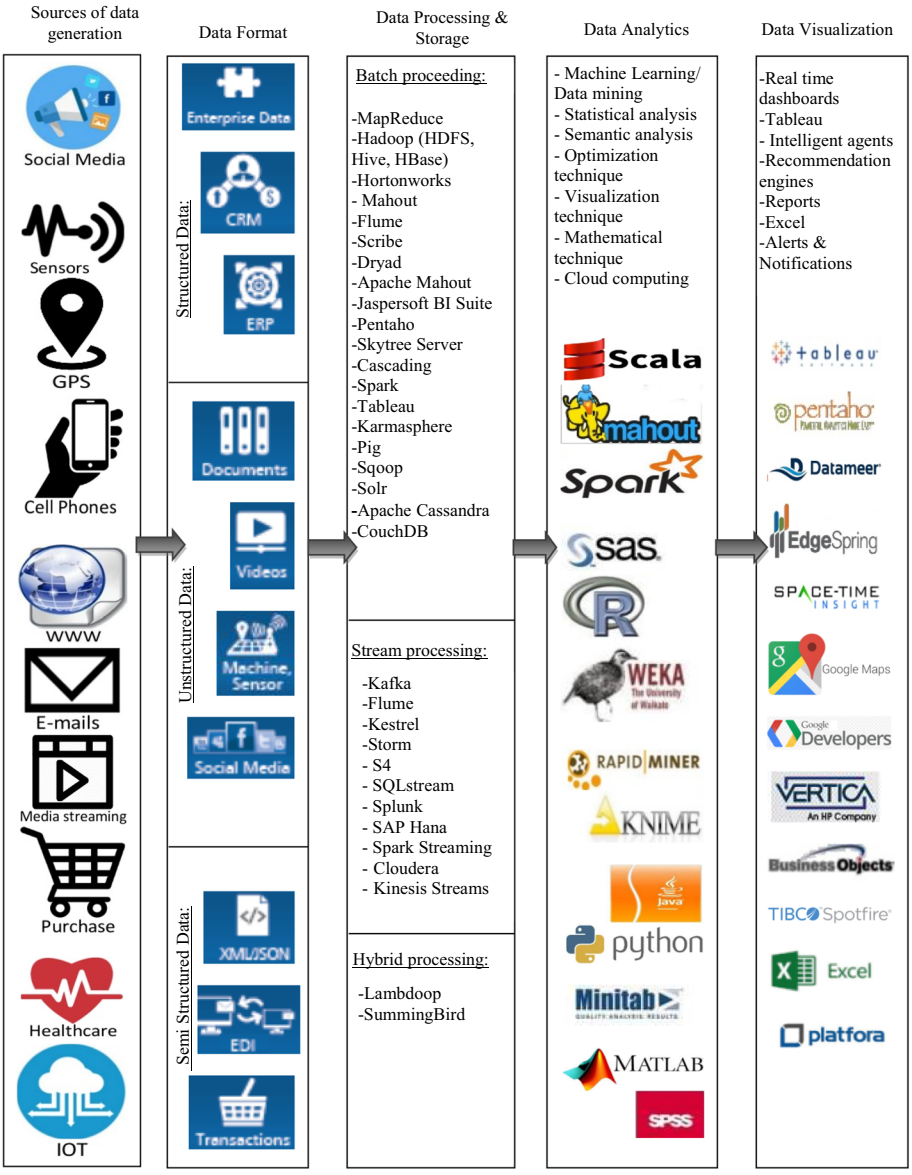

Fig. 3 Classification big data Framework

analytics including machine learning/data mining, fuzzy sets, statistical analysis or mathematical techniques, database querying, visualization techniques, semantic analysis, optimization methods. Our classification framework for big data research papers is presented in Fig. 3.

4.1 Classification framework for domain areas of big data and data sources

4.1.1 Domain area

Many big data platforms have been designed to provide users or organizations with big data applications to help them on pollution control, traffic management, disaster management, social management, health and safety in rapid growth of data from every imaginable source such as Internet, purchase transactions, sensors, and social media networks (Kousiouris et al. 2018; Kovalchuk et al. 2018; Huang et al. 2017). However, finding articles that classify big data research articles systematically is not easy, even though bid data have been employed to diverse commerce and business areas. Hence, it is meaningful that articles is reviewed and classified by domain areas or big data applications. We classify articles by applications of the Big Data in: (1) Health and medical care, (2) Social networking and internet, (3) Government and public sector, (4) Natural resource management, (5) Economic and business sector. Through in-depth reviews of articles, classifying areas of government and public sector involves smart city, transportation and traffic management, network security and classifying natural resource management involves water resources management, energy and utilities management. Also, finance and fraud detection and telecommunications fields will be considered in economic and business sector. Following paragraphs summarize some recent contributions in big data applications.

1. Health and medical care

Big data analytics is commonly applied in many areas and one of the areas is the health and medical care. This is due to the large amount of and growing dataset that this area has. In 2011, it was estimated that health and medical care's clinical data stood at approximately 150 Exabytes, climbing at the rate of between 1.2 and 2.4 Exabytes per year.[1] Clinical data comes from the electronic medical records (EMRs) and imaging data. Usually, this data is rapidly increasing both in size of records and coverage in population. EMRs usually have personal data that is on a person's genetic and DNA, and data from other sources such as medication records (Manogaran et al. 2017; Nguyen et al. 2017). Other than clinical data, healthcare data also includes pharmaceutical data, data on personal practices, personal preferences (patients' habits, dietary patterns, surrounding factors, etc.) and financial/activity records. The integration of these data sets is important since it would help in coming out with effective intervention program or ensuring patients' well-being in getting their treatments. One of the most taxing challenges in integrating these data is doing the large-scale integration and analysis (Nair et al. 2017). Virtually, data is collected at point-of-care and is stored in repositories. To illustrate, imaging data (MRI, fMRI) is often accessed by trained radiologists overseas or from other hospitals or healthcare centre to provide expert opinion and diagnoses. Even if the format in data interchange is fully standardized, the multimodal nature of data is still a taxing challenge in the integration of the data. Other than the integration of the data, the analysis is also a taxing challenge. This is due to the influence from other data gathered such as those from high-throughput screens such as patient diagnoses and progression of outbreaks as well as the time factor. Issues like quality of data, privacy, security and the effectiveness of the analysis are crucial in

[1] http://blogs.sas.com/content/hls/2011/10/21/how-big-is-big-data-in-healthcare/

healthcare informatics. Clinical data management must be within the perimeter of established law and practices and also in lieu of a person's or a patient's expectation of privacy. Another issue is intellectual property which is evident in pharmaceutical data. It must be guarded properly especially in a collaborative environment. In recent years, healthcare IT has gained significant investments especially in terms of computing devices and it is expected to increase sharply in future due to its impact onto the growth of the medical and healthcare area (Batarseh and Latif 2016; Manogaran et al. 2017).

The attention of many researchers has been elicited in healthcare science applications by the advent of big-data technology. These applications usually involve sources of data streams generated by a large number of distributed sensors. Data are collected from the real-time signals of blood glucose temperature, blood pressure, respiratory and heart rate, cardiovascular status and chest sound. Then, such data are sent to a mobile device and backend data center for processing and medical treatment (Manogaran et al. 2017; Zhang et al. 2015).

Big data analytics play a vital role in the process of disease exploration and care delivery. These analytics have been recently utilized towards aiding in medical decision support system in the healthcare practice. However, computational scientists come up with innovative solutions by considering the exponential growth of the amount of medical to process this large volume of data in tractable timescales.

2. Social networking and internet

In recent years, we have seen many social media like LinkedIn, Twitter, and Facebook that have exploited big data; being used to disseminate information to users. For example, when a 5.9 Richter earthquake hit near Richmond, VA, on August 23rd, 2011, residents in New York City read about the quake on Twitter feeds 30 s before they experienced the quake themselves. This vividly describes the speed and impact of information flow in social networks. In its form which is unstructured, one can expect a significant rise in the personalization and real-time delivery of content to the users. This is apparently true in the evolution of search engines and it is one of the most successful big data analytics applications (Nguyen et al. 2017; Vennila and Kannan 2016; Plimpton and Shead 2014). The Internet itself operates as the infrastructure of a semantic web. Since most of its information is currently unstructured, there is a significant motivation for organizing it into structured forms to infer related concepts and relations, in order to mechanize reasoning. This is crucial in a hypertext media like the social media in which is used not only to disseminate information but also to market a service or a product. Analyzing the interactions of data as in the functions of time and other attributes will help to understand the emergence of patterns of collective behaviors, resource management, enable the shaping of information flows, and make prediction. In order to this, the summary of link representations for marking substructures or nodes of interest which present targets for online marketing strategies is needed. In addition, the novel graph mining technique is also needed although it is relatively in its infancy stage. Text search and indexing technologies which are relatively mature are also crucial since they provide new dimension in the search for entity correlations, putting additional strain on big-data analytics infrastructure (Ahmad et al. 2017; Hidalgo et al. 2017).

Castiglione et al. (2017) proposed a big data infrastructure to deal with digital contents in cultural heritage environments such as multimedia contents and social data. In this research, the application of a scalable cloud-based infrastructure and context-driven

analysis for big data resource management will be shown in cultural heritage domain. Castiglione et al. (2017) presented a big data infrastructure to query, analyze and process digital cultural contents from heterogeneous and distributed sources (e.g., Digital Libraries, Data Services, Social Media, Sensor Networks, Web Multimedia Collections, etc.) and then, they present needs of users in a suitable format. They provide the information retrieval facilities, with application of context-awareness in data access and analytics based on preferences of users and on the surrounding environment.

The extensive use of social network like Weibo, Twitter, and Facebook has caused and is causing the production of large volume of data. The commercial and non-commercial organizations are showing their keen interest in discovering new business insights which would help them to increase their performance. By using advanced analytics, these organizations can analyze big data to learn about the relationships that exist among the social networks. These networks characterize the social behaviors of individuals and groups. By utilizing the data to describe the relationships, we are able to identify social leaders who influence the behaviors of others in the network. This also enables us to determine the people who are the most affected by other network participants.

3. Government and public sector

Both public and private sectors have encountered problems with big data. This is due to the size of population which is huge and the population is from different age groups and each has different needs when it comes to public services. Each person in a community produces a lot of data in each public sections, hence making the data in public administration becomes enormous. As it is commonly known, a government is responsible in managing the computerized system in transportation, social welfare, as well as safeguarding sensitive data and the system from any malicious cyber-attack. Xia et al. (2016) proposed a framework for traffic data processing to analyze data in intelligent monitoring and recording system for monitoring traffic in cities and vehicle trajectory tracking. Many governments are focusing on building the modern era such as Smart City applications in urban area as a major impact on the life of citizens. The Smart City uses electronic data collection sensors and connected devices over Internet in order to monitor all the city and information about citizen, and their daily routine and houses. It facilitates the citizen regarding pollution control, disaster management, safe from theft, assaults, and health, and so on. The responsibility of the Smart City is to provide good sanitation system, a clean environment, a lot of parking areas at proper places, cycling tracks, and safety from natural disasters, such as earthquakes, heavy rains, tsunami, floods, and thunderstorms. Even the city has the technology to detect disaster happens for people when any disaster happens, and take precautionary measures to save the people' life (Kousiouris et al. 2018; Jiang et al. 2018; Rathore et al. 2017; Liang et al. 2016).

Another significant analytics application in big data is fraud detection. The adoption of electronic payment has increased and this has led to new perspectives among scammers. Hence, innovative countermeasures are needed to deter these criminals' scam. They are continuously improving their modus operandi and simultaneously, companies that are managing transactional services should also be collecting data and monitoring customers' purchasing behaviour. The expansion of electronic commerce and the increasing confidence among customers in electronic payment has cause fraud detection to become more significant than ever. Fraud detection needs a meticulous design and

technique so that its implementation and analysis is compatible with enormous amount of streaming data (Wang and Belhassena 2017).

The public sector is turning into increasingly more conscious of the potential value to be gained from huge information, as governments generate and collect great quantities of records through their day-to-day activities. As result, the benefits of huge data in the public sector can be grouped into three essential areas: through automatic algorithms; enhancements in effectiveness, supplying increased internal transparency; enhancements in efficiency, where better offerings can be supplied primarily based on the personalization of services; and learning from the overall performance of such services.

4. Natural resource management

The changes that the world is going through has affected our habitats and impacted the environment in the long term. Hence, there has been much data on our environment and the impacts that they have onto our life being collected. Advancement in data computing has resulted in images captured on screens about our nature, for example, at high spatiotemporal resolution, nature calamities such as deforestation and urban encroachment can be observable in high definition. In addition, we can also monitor receding glacial ice caps, melting of icebergs and extreme weather occurrences. This data is commonly collected from satellite images, weather radar, and terrestrial monitoring and sensing devices. In recent years, efforts have been taken on collecting the targeted data on the carbon footprint of key sources. These datasets are typically found in larger data centers that offer relatively high-throughput access. Other big data problems include analyzing data in natural resource management, including water and land resources management, sustainable development, and environmental impact assessment. Data for the analysis come from varied sources including sensors monitoring environmental state, human activities such as in manufacturing or agricultural and other factors. Even though the analysis is still at its early stage, such model of analysis is critical in sustainability (Huang et al. 2017; Yuan et al. 2017; Ahmad et al. 2016).

As result, the term “climate change” can cover some natural and some manmade, including loss of wildlife habitat and global warming. Each of those brings its own challenges however, progressively; massive knowledge and analytics are being place to use to return up with new solutions and analysis ways. Global climate change has been attracting loads of attention for an extended time because of the adverse effects of it is being felt everyplace.

5. Economic and business sector

The most noticeable application of big-data analytics has been in the commercial sector. It is estimated that if a retailer is fully utilizing the power of analytics, its operating margin can be increased by 60%. A comprehensive analytics framework requires the integration of supply chain management, after-sales support, customer management, advertising, etc. Business enterprises collect immense amounts of multi-modal data, including inventory management, store-based video feeds, sales management infrastructure, advertising and customer relations, customer transactions, customer preferences and sentiments, and financial data. The combination of all data for large retailers is easily estimated to be in the Exabytes, currently. Datasets in such applications are relatively well structured and integrated. Since these analyses typically operate in closed systems (in which much of the

data, analyses and infrastructure are performed within the same security domain), issues of privacy and security in analyses are more manageable. Data quality is not a major concern and resources are available in the state-of-the-art data centers. The major bottleneck in this domain is the development of novel analytics methods which are compatible with the vast amounts of multimodal data (Ding et al. 2017; Singh and Bawa 2017; Wang et al. 2017a, b).

Noted that, the opportunities of big data analytics in business have been a good deal discussed in latest years, but this is additionally still a rising area, with many uncertainties about what enterprise fashions will succeed

4.1.2 Source of data

A rapid growth of data comes from every imaginable source such as Internet, social media networks, Sensors, cultural items' descriptions (Wang and Belhassena 2017; Spivak et al. 2018; Prajapati et al. 2017; Plimpton and Shead 2014).

- *Internet* The rich sources of data generation are the internet. There are devices such as GPS devises, smart phones, and window sensors which on connected through the internet offers different kind of services that produced huge amount of crude, largely unstructured data.
- *Social media/multimedia data* Huge amount of data being generated through user comments and opinions from the social sites such as Twitter, Facebook, LinkedIn, Microblogs, Blogs. Also, image, text, video, and audio retrieved from open networks (Social Media Networks as YouTube, Picasa, Flickr) or private collections;
- *Sensors* The sensor networks produce a huge amount of data and are used in different types of systems such as cultural environment, weather research and forecasting.
- *Web service data* Kind of data collected by means of web services or several digital libraries. For example, descriptions of cultural items are coming from open web sources (e.g., Wikipedia).
- *Other devices.* There are other data which may not consider as aggressive as social media data, but their footprints cannot be ignored. Example includes power meters.

4.1.3 Format of data

The variety formats of data in big data life are as follows: (Arias et al. 2017; Ding et al. 2017; Singh and Bawa 2017; Nair et al. 2017)

- *Structured data* The structured data have a specific format of data and has a relational structure. The management of such data is easily possible using a standard SQL-type language, often seen in relational database management systems. Examples of structured data are string, numeral, dates, etc.
- *Unstructured data* The unstructured data does not follow any specific format. Data generated in different format such as videos, text, time information and geographic location. The rapid development and management of such data poses severe challenges to the existing computational capacity.
- *Semi-structured* Managing the semi-structured data is not possible by traditional database management systems techniques, but the understanding and analysis of such data

is not impossible either. Some comprehensive and intelligent rules are required to dynamically decide the next process, once the previous process is over.

4.2 Classification framework for big data tools in data processing and storage

There are three main tools for big data computing, namely, batch processing tools, stream processing tools, and hybrid processing tools. Most batch data analytic platforms are based on the Apache Hadoop, such as Dryad and Mahout. Examples for streaming data analytic platforms are Storm and S4 to be used in real time applications. Hybrid processing tools takes advantages of both batch processing and stream processing for computing massive quantities of data. The following sub-sections discuss big data technologies classified by distinct processing tools.

4.2.1 Batch processing

Batch processing models and transforms the Data Lake's files into Batch Views ready for the analytical use-cases. It is responsible to schedule and execute Batch Iterative Algorithms, such as sorting, searching, indexing or more complex algorithms such as PageRank, Bayesian classification or genetic algorithms. Batch Processing is mostly represented by the MapReduce programming model. Its drawbacks appear twofold. On one hand, when processing huge amounts of batch data, several jobs may usually need to be chained so that more complex processing can be executed as a single one. On the other hand, intermediate results from Map to Reduce phases are physically stored in hard disk, completely detracting the Velocity (in terms of response time). Massive efforts are currently put on designing new solutions to overcome the issues posed by MapReduce. For instance, by natively including other more atomic relational algebra operations, connected by means of a directed acyclic graph; or by keeping intermediate results in main memory (Genuer et al. 2017; Kumar and Rath 2015). We discuss the most common tools based on batch processing in the following paragraphs:

- Hadoop/MapReduce

Apache Hadoop is well known example of a batch processing framework to support the distributed storage and processing of large sets of data on clusters of commodity hardware. It is an open-source Java-based framework which is widely used by large corporations like Facebook, Yahoo! Twitter to store and process Big Data workloads.

In fact, Hadoop consists of two components: (1) the Hadoop Distributed File System (HDFS) in which the storage of data among nodes of a cluster is distributed; and (2) the Hadoop MapReduce engine which assigns the data processing to the node where it resides. Hadoop is an open source implementation of MapReduce programming model for processing large datasets, structured as in a database or unstructured as in a file system by using parallel and distributed algorithm on a cluster of nodes consists of one master node and multiple worker nodes. The master node takes the input, divides input into smaller sub-problems, and distributes these smaller sub-problems to worker nodes. A worker node may do this process again in turn, leading to a multi-level tree structure. The worker node passes the answer back to its master node after processing the smaller sub-problem. Then, the answers to all the sub-problems are collected by the master node and these answers are combined in some way in order to form the output of answer to the problem that the

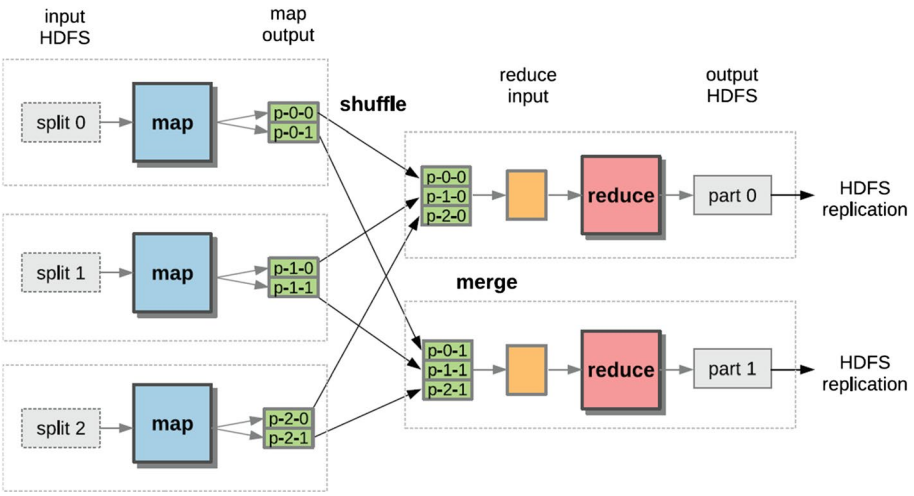

Fig. 4 MapReduce scheme performed by Hadoop

master node was originally trying to solve (Elkano et al. 2017; Kranjc et al. 2017). Hadoop is also the base framework of Apache Mahout[2] which has machine learning algorithms to support dimensionality reduction, recommendation mining, frequent item-set mining, classification, clustering. The first versions of Mahout implemented the algorithms built on the Hadoop framework, but recent versions of Mahout include new implementations which run on Spark for example implementations of Spark-item similarity enable the next generation of co-occurrence recommenders in which user click streams and contexts in making recommendations is used. However, Hadoop features authentication, load balancing, high availability, flexible access, scalability, tunable replication, fault tolerance, and security for Big Data applications including financial analysis, machine learning, natural language processing, genetic algorithms, simulation, and signal processing and so on. Due to problems such as lack of scalability in some components in Hadoop, the second version of Hadoop with appearance of Yet Another Resource Negotiator (YARN) was caused. YARN is a resource management framework which splits the responsibilities of job tracker and task tracker in MapReduce and thus multiple applications can be run in parallel while sharing a common cluster resource management. Hadoop clusters in new engines can be scaled up to a much larger configuration and support iterative processing, stream processing, graph processing, and general cluster computing all at the same time. An overview of overall MapReduce scheme performed by Hadoop is shown in Fig. 4. At the first step, the input data is divided into splits and splits are read by the map functions in (key, value) format. The output of map is partitioned into different fragments (p-x-x in Fig. 4). Shuffle step redistributes fragments produced by the map function, such that all data belonging to same behavior (key) is located on the same node. Then, reduce step used as a combiner and run on map outputs. The corresponding reducers process their input to merge them and generate the final results and sent results to HDFS (Najafabadi et al. 2017a, b; Pedersen and Bongo 2017; Manogaran et al. 2017; Xia et al. 2016).

[2] https://mahout.apache.org/.

HBase and Hive are a part of the Hadoop framework, which built on top of it. HBase is a scalable distributed storage system to solve the big data processing problem that traditional relational database faces today. The Hive is a robust data warehouse platform to managing and querying the distributed Big Data sets. Hive comes equipped with a SQL-like query language, called HiveQL. Hive performs equally well with any data type—such as user-created formats, control delimited, without any reservation. Hive is not competent enough for real-time query processing but outperforms its peers for batch jobs on append-only type Big Data such as web logs (Manco et al. 2017).

- Apache Pig

Apache Pig is an integral component of Hadoop ecosystem to reduce the data analysis problems with executing data flows in parallel on Hadoop. Pig is a structured query language (SQL) which is being used by large organizations like LinkedIn, Twitter, Yahoo, etc. The scripting language for this platform is called Pig Latin which abstracts the programming complexity in MapReduce from other languages such as Java into high level notations. Pig is one most complete platform, because it can invoke code in many languages like JavaScript, Java, Jython and JRuby through direct calling to the User Defined Functions (UDF). Hence developers can do all required data manipulations in Hadoop with Pig. Pig can be used as a component with considerable parallelization to build complex and heavy applications that tackle real business problems with their Big Data sets. In a typical scenario, Pig with the help of UDFs works with data from files, streams, structured and unstructured data and performs the operations such as select, transformation or iteration and finally stores the results into the HDFS (Manco et al. 2017; Manogaran et al. 2017).

- Flume

Flume is employed as the instrument to feed data into Hadoop. Together with a processing framework, a message passing layer is needed to access and move streaming data. Apache Flume is one of the more mature options that offer this. Flume has been a renowned application for data feeding. It is well-embedded in the overall Hadoop ecosystem and gains support from all the commercial Hadoop distributions. This has caused Flume to be the main choice among the entreprenuers. In addition, Flume is always compatible with new Hadoop products. However, it also has a setback. It tends to lose things occasionally due to unavailability of event replication (Bharti et al. 2016).

4.2.2 Stream processing

Processing enormous amount of data in parallel is not a problem to Hadoop. It is a general dividing mechanism to distribute accumulation of workload across different machines. In addition, Hadoop is designed for batch processing. Hadoop is a multi-purpose engine but not a real-time and high performance engine due to its latency. In some stream data application like log files processing, machine-to- machine, sensory industry and telematics require real-time response to process large stream data. Hence, it is necessary to have real-time analytics for stream processing. Streaming big data requires real-time analytics since big data has high velocity, high volume, and complex data types to be developed. In applications that involve real-time processing, there are challenges to Map/Reduce framework when time dimension and high velocity are concerned. Therefore, the real-time Big Data

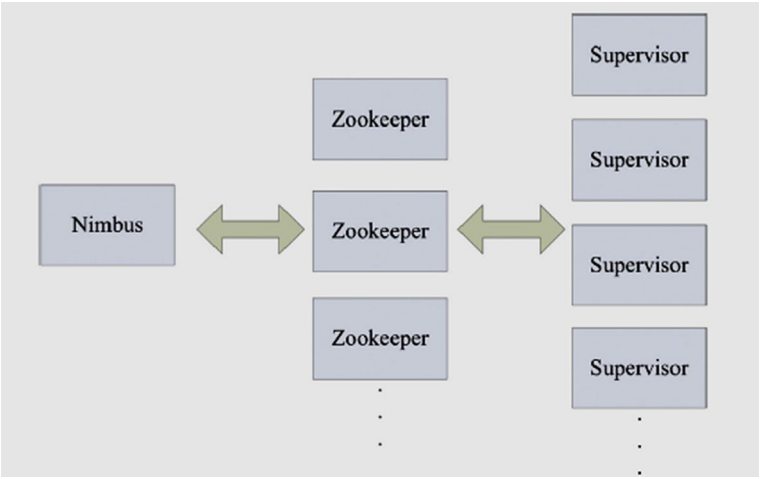

Fig. 5 A storm cluster

platforms as stream processing such as Storm, S4, Splunk and Apache Kafka have been developed as the second generation for data stream processing in real- time analysis of the data (Wang and Belhassena 2017; Ferranti et al. 2017). Real-time processing means that the continuous data processing highly needs an extremely low latency of response. This is due to the small volume of accumulated data at the time dimension of the processing. Generally, big data may be collected and stored in a scattered environment, not in one data center. Usually, in the Map/Reduce framework, the Reduce phase only starts to work post the Map phase. Therefore, all the intermediate data generated in Map phase is saved in the disk before being submitted to the reducers for the next phase. All these lead to significant hindrance of the processing. The high latency characteristic of Hadoop makes it almost impossible for real-time analytics. The most common tools based on stream processing are explained in the following paragraphs:

- Storm

Storm[3] is one of the most recognized programs for data stream processing in real-time analytics which concentrates on assured message processing. Storm is a free and open source distributed streaming processing environment for developing and running distributed programs that process constant torrent of data. Hence, it can be said that Storm is an open source, general-purpose, distributed, scalable and partially fault-tolerant platform that reliably process unbounded streams of data for real-time processing.

One advantage of developing Storm is that permits developers to focus on using a stable distributed process while delegating the complexity of distributed/parallel processing and technical challenges like the construction of an elaborate recovery mechanism to the framework. Storm which is an intricate event processor and distributed computation framework, is written basically in the Clojure programming language. It is a distributed real-time computation system for rapid processing of large streams of

[3] http://storm.incubator.apache.org/.

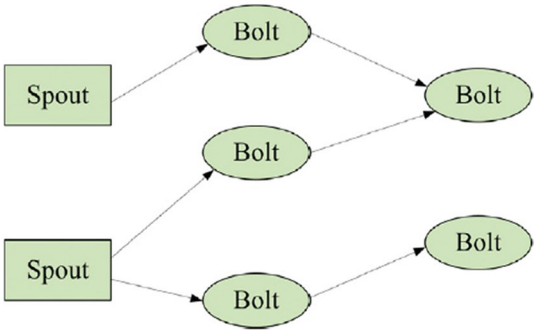

Fig. 6 Example of a storm topology

data. Storm is a distributed/parallel framework consisting of Nimbus, Supervisor, and Zookeeper as illustrated in Fig. 5. Storm cluster mainly consists of master and worker nodes, with coordination done by Zookeeper (Wang et al. 2017a, b; Um et al. 2016; Agerri et al. 2015).

First, Nimbus, in Storm, distributes codes to perform parallel processing, delegates tasks to Supervisor, and handles error. Second, Supervisor performs the role of initiating a work process to handle the Topology created to process multifaceted events. Topology nodes fall into two categories which are known as Spout and Bolt as shown in Fig. 6, with Spout acting as an event receiver that collects streaming data or events from many sources and delivers data to the multiple Bolts. Bolts then perform the roles of unit logic event processor such as filtering, collecting, and joining for event flow on the event processing network. Lastly, Zookeeper as coordination service for distributed applications is responsible for synchronizing the nodes and acts as a distributed coordinator to coordinate the system and records all situations of Nimbus and Supervisors on local disk. It supports a resurgence device against errors to ensure that the framework is fault-tolerance (Karunaratne et al. 2017; De Maio et al. 2017).

The main abstraction structure of Storm is the topology. It is a top level abstraction which describes the processing node that each message passes through. The topology is represented as a graph where nodes are the processing components, while edges represent the messages sent between them. Topology nodes are spout and bolt nodes. Spout nodes are entry points of a topology and source of initial messages to be processed. Bolt nodes are the actual processing units, which receive incoming text, process it, and pass it to the next stage in the topology. Several instances of a node in the topology can allow actual parallel processing. The data model of Storm is a tuple. Each bolt node in the topology consumes and produces tuples. The tuple abstraction is universal enough to allow any data to be passed around the topology. In Storm, each node of the topology may exist in a different physical machine. Nimbus that is Storm controller is responsible for distributing the tuples among the different machines, and ensuring that each message traverses all the nodes in the topology. In addition, Nimbus performs automatic rebalancing to compensate the processing load between the nodes. Storm integrates with the queueing and database technologies that are already used. A Storm topology consumes streams of data and processes those streams in arbitrarily intricate ways. However, repartitioning the streams between each stage of the computation is needed.

- S4

The S4 is a fully distributed stream processing platform inspired by the MapReduce model. The stream operators are specified by the user code and the configuration jobs described with XML. S4 is a general-purpose, fault-tolerant, scalable, distributed, pluggable computing framework that programmers can easily develop applications for processing continuous unbounded streams of data. It was initially released by Yahoo! in 2010 and has become an Apache Incubator project since 2011. S4 allows programmers to develop applications based on several competitive properties, which included scalability, decentralization, robustness, extensibility and cluster management. The S4 is written in Java. The tasks of S4 job are to be modular and pluggable for easy and dynamic processing of large-scale stream data. S4 uses Apache ZooKeeper to manage its cluster, like Storm does. S4 is used in developing systems at Yahoo! for processing thousands of search queries. Although S4 is an effective analytics framework for real tile event analytics, it has the potential to cause message loss. This is due to its recovery device that is based on the checkpoint method. The recovery device is used to support fault tolerance (Chen and Zhang 2014).

- Kafka

Kafka is an open-source distributed streaming framework that was initially developed by LinkedIn in 2010. It is a flexible publish subscribe messaging system that is designed to be speedy, scalable and commonly used for log collection. Kafka is written in Scala and Java. It has a multi-producers management system that is able to salvage messages from multiple sources. For testing purposes, the streaming was emulated through a bash program by injecting transactions in Kafka at a desired rate per second. Generally, Kafka's data partitioning and retention make it a useful tool for fault tolerant transaction collection. This is because applications can develop and subscribe to streams of records, with fault tolerance guarantees and the possibility of processing streams as they occur. Since Kafka does not use HTTP for ingestion, it delivers better performance and scale. Similar to other publish-subscribe messaging systems, Kafka stores streams of records in categories called topics, and each record is basically made up of a key-value pair with a timestamp. Kafka is a tool to cope with streaming and operational data via in-memory analytical techniques for obtaining real-time decision making. Kafka as a distributed messaging system has four main attributes: high-throughput, persistent messaging, support for distributed processing and for parallel data load into Hadoop. It already has wide practices in a number of various companies as messaging tools and data pipelines.

In recent years, activities and operational data play a crucial role in extracting features of websites. Activity data is record of different people's activities on line, such as webpage content, click-list, copy content, and searching key words. It is meaningful to log these activities out into canned file and aggregate them for subsequent analysis. Operational data describes the performance of servers, for example, CPU and IO usage, service logs, request times, etc. The knowledge detection of operational data is beneficial for real-time operation management. Kafka combines off-line and on-line processing to develop real-time computation and provide impromptu solution for these two kinds of data. Kafka uses Zookeeper as a distributed coordinator and Topic consumer offset manager to keep a parallel distributed coordinated file structure for parallel low latency data access, with single write operations directed to a 'leader node'. Zookeeper is responsible for forcing data propagation down to the available nodes for distribution (Tennant et al. 2017; Castiglione et al. 2017).

- Flink

Flink is a dispersed processing component that concentrates on streaming processing, which has been designed to solve problems derived from micro-batch models (Spark Streaming). Flink also supports batch data processing with programming abstractions in Scala and Java, though it is treated as a special case of streaming processing. In Flink, every job is executed as a stream computation, and every task is performed as cyclic data flow with several iterations. Flink provides two operators for iterations which are the standard and delta iterator. In standard iterator, Flink only works with a single partial solution and in delta iterator, Flink employs two work sets: the next entry set to process and the solution set. Among the benefits of iterators is the reduction of data to be computed and sent between nodes. New iterators are specially designed to tackle machine learning and data mining problems. Apart from iterators, Flink influences an optimizer that analyzes the code and the data access conflicts to reorder operators and create semantically equivalent execution plans. Physical optimization is then applied on plans to enhance data transport and operators' execution on nodes. Finally, the optimizer selects the most resource efficient plan concerning network and storage. In addition, Flink also provides a complex fault tolerance mechanism to consistently recover the state of data streaming applications. This mechanism produces consistent snapshots of the distributed data stream and operator state. In case of failure, the system can fall back to these snapshots. Aim of FlinkML is to provide a set of scalable machine learning algorithms and an intuitive API to Flink users. FlinkML has been provided several alternatives like Multiple Linear regression or SVM for supervised learning, k-NN join for unsupervised learning, scalers and polynomial features for preprocessing, Alternating Least Squares for recommendation, and other utilities for validation and outlier selection in some areas in machine learning. FlinkML also allows users to build complex analysis pipelines via chaining operations (like in MLlib from Apache Spark).

- Spark

A more current alternative to Hadoop is Apache Spark. It includes an extra component called MLlib that is a library geared towards machine learning algorithms such as: clustering, classification, regression, and even data preprocessing. Due to capacity of Spark, batch and streaming analysis can be done in the same platform. Spark was developed to overcome Hadoop's weakness that it is not optimized for iterative algorithms and interactive data analysis, which performs multiple operations on the same set of data. Spark is defined as a the next generation of distributed computing frameworks which can process large volume data sets in memory with a quick response time due to its memory-intensive scheme (Oneto et al. 2017; Wang et al. 2016).

Spark also perform query processing and evaluation on big data which is useful in optimizing huge data management workflow, by developing a high level Application Programming Interface that introduce significant effects on productivity in applications development. Applications can request distributed processing operations such as map, reduce and filter by passing specific closures (i.e., functions) to the Spark runtime framework. The heart of Spark is formed by Resilient Distributed Datasets (RDDs) which controls the distribution and transformation of data across the cluster. Users define the high level functions or additional operations over the data without strictly sticking to map and reduce functions. RDDs consist of a collection of data partitions distributed across several data nodes (Tsai et al. 2017; Nguyen et al. 2017). A wide range of operations like filtering, grouping and

operation setting are provided for transforming RDDs. Furthermore RDDs are also highly adaptable as they allow users to customize partitioning for an optimized data placement, or to preserve data in several formats and contexts. RDDs can be considered as lazy in the sense that they will only compute an action when they are invoked. Hence, an application is needed to overcome this weakness. The application can be implemented as a series of actions on the RDDs. This is because when an action is executed over RDDs, a job will trigger. In other word, Apache Spark is a distributed computing system that is based on the concept of RDDs. RDDs are series of elements that can be operated in parallel on the nodes of a cluster by using two types of operations which are transformations (map, filter, union, etc.) and actions (reduce, collect, count, etc.). The unique features of Spark are it is a high level parallel processing programming model, a graph processing, machine learning algorithms, multi-programming language API, it can run on different systems (Mesos, standalone, Hadoop, cluster), and it can do streaming processing. Spark is becoming highly popular and is replacing MapReduce as the dominant technology for developing Big Data applications. To solve fault tolerance in Spark, the operations are interpreted in a structure known as lineage. The transformations annotated in lineage are only performed when I/O operations occur in the log. Should a failure occurs, Spark will re-compute the affected one in the lineage log. Spark allows data in local disk to be spilled should the memory capacity is not sufficient. Developers of Spark have come out with another abstraction that is of high level that is called Data-Frames. This leads to the concept of formal schema in RDDs. Data-Frames is a collection of structured and distributed data organized by identified columns. This is a constant relational database or data frame in R, or Python. In addition, the relational query plan developed by Data-Frames is optimized by Spark's Catalyst optimizer via the defined schema. This has caused Spark to comprehend the data and remove the expensive Java serialization actions (Tsai et al. 2017; Nghiem and Figueira 2016).

4.2.3 Hybrid processing

The hybrid processing enables the possibility of coming big data platforms into the third generation as it necessary for many domains in big data applications. This paradigm synthesizes both batch processing and stream processing paradigms based on the Lambda Architecture. Lambda architecture is a data-processing architecture designed to handle massive quantities of data by taking advantage of both batch- and stream-processing methods. A high-level architecture of this paradigm contains three layers. Batch layer manages the master dataset that has been stored in a distributed system and is not changeable. Serving layer load and exposes the views of batch layer in a data store for query, and speed layer only deals with new data with low latency. At last, a complete result is merged by the combination of batch and real-time views (Wang et al. 2016; Chen and Zhang 2014).

5 Classification framework for big data analytics techniques

In general, scientists have developed a wide variety of big data techniques to analyze and extract knowledge from large amounts of data within a limited time period. These techniques have been applied for the exploration and analysis of large quantities of data and used for many data-intensive applications in order to discover meaningful patterns and rules. They can be used to show their effectiveness for capturing, curating, analyzing and visualizing Big Data and their significance in decision making. However, some researchers

have used big data techniques to model past behaviour based on historical data or descriptive analytics (Arias et al. 2017; Oneto et al. 2017), other researchers attempt to predict the future by analyzing current and historical data or assist analysts in decision making by assessing actions regarding business objectives, requirements, and constraints (Mavridis and Karatza 2017; Nair et al. 2017). In this section, we review the current trends of big data techniques in following our classification framework (Fig. 3 at Sect. 4). We widely classified techniques used in big data analytics into the following six categories: machine learning/data mining techniques, cloud computing, semantic network analysis/web mining, visualization techniques, mathematical and statistical techniques, and optimization techniques. The following sub-sections describe these big data techniques: these techniques are the most of which have shown their capabilities in processing Big Data

- Machine learning/Data mining techniques (deep learning/artificial neural networks)

Data mining and machine learning techniques are a set of artificial intelligent techniques to extract hiding knowledge and valuable information (patterns) from data once a designed algorithm learns behaviors from empirical data. The algorithms include support vector machine, cluster analysis, classification, association rule of learning, and regression. New techniques based on big data architectures are required in managing and analyzing big data because traditional data mining techniques are efficient in analyzing data but not efficient and scalable to cope with big data. Data rate is increasing rapidly, therefore k-means, fuzzy c-means, hierarchical clustering, clustering using hierarchies, CLARANS, balanced iterative reducing and clustering large applications should be extended for the future use of data clustering, so that they can cope with the huge workloads; otherwise, these algorithms would no longer be applicable in the future (Najafabadi et al. 2017a, b; Wang and Belhassena 2017; Wang et al. 2017a, b). Parallel programming model, such as Hadoop and Map/Reduce can scale up data mining and machine learning techniques for mining and parallel processing of big data sets. For example, artificial neural network (ANN) which is fundamental algorithm for image analysis, adaptive control, pattern recognition, and so on, suffer from time and memory consuming in learning process when Big Data is concerned. The more hidden layers and nodes are needed in a neural network for higher performance. However, ANN will be resulted in poor performance and extra time consumption over Big Data because of its complexity in learning process. Thus, ANNs have been improved in a parallel and distributed setting to reduce memory and time consumption (Fonseca and Cabral 2017; Rahman et al. 2016). Deep-Learning technique is one of the popular techniques using ANN to extract information from complex datasets and to discover correlations from data. It makes the sense of different types of data such as images, text, and sound by learning multiple levels of representation and abstraction. Deep Learning have played important role in many Big Data application including pattern recognition, text mining, image analysis, adaptive control and genomic medicine. Most of the big data analytics techniques are based on deep-learning approach that employs classification optimization, statistical estimations, and control theory in solving Big Data analytics problems. The learning process of large-scale data sets by a neural network requires large memory due to its needs for more hidden layers and nodes to produce higher accuracy. It is generally acknowledged that neural processing of big data leads to very large networks. In fact, one of the main challenges in this process is memory limitations and training time that are increasingly intractable. In addition, the conventional training algorithms also perform poorly. Therefore, some sampling approaches can be employed to reduce the size of data and to scale up neural networks in parallel and distributed ways (Chen and Zhang 2014;

Iqbal et al. 2017). Deep learning also showed its efficiency for new types of recommendations, such as cross-domain recommendation systems in which items are mapped to a joint latent space, and the social trust ensemble learning model (Iqbal et al. 2017; Najafabadi et al. 2017a, b; Najafabadi and Mahrin 2016).

Consequently, it is necessary to redesign the machine learning and data mining algorithms on MapReduce framework in order to mine large scale of data (Zhang et al. 2016). Mahout and Spark MLlib are two open source projects that tackle scalability problems and support many algorithms including regression, classification, and collaborative filtering, clustering and dimensionality reduction. Because they provide a distributed environment in which algorithms can process large datasets.

- Cloud computing technologies

Cloud computing is emerging as a crucial resource in efficient data processing and it has become the main shift in recent the computational information age that promises an reasonable and dynamic computational architecture for large-scale and intricate enterprise applications. It is an important and revolutionary model that offers service-oriented computing and abstracts the software-equipped hardware infrastructure from clients or users. The main goal of cloud computing in the Big Data application is to explore the large quantity of data and extract useful information or knowledge for future actions. Cloud computing is a parallel distributed computing system used in the Big Data analytics. Cloud computing services acknowledge users' requests for information sharing then make the best decisions from the data and pass the information to other users without redundancy. The concept of cloud computing is mainly popular in three aspects: (1) Software-as-a-Service (SaaS), (2) Platform-as-a-Service (PaaS) and (3) Infrastructure-as-a-Service (IaaS). Briefly, SaaS provides users with uninterrupted access to applications. PaaS helps users to develop, run, and manage applications. IaaS offers users with access to a pool of configurable computing resources like network, storage and servers. Cloud providers own and operate cloud services. A cloud service operated for a single organization is called a private cloud, and the one that operates for public use is called a public cloud. As the era of big data and cloud computing is coming, huge number of data centers have been widely deployed around world and thus power demand of the global data center increases rapidly. In 2013, the energy consumption of data centers accounted for 0.5% of the world total energy consumption. It is estimated that, the energy consumption of data centers will increase in 2020 (will account for 1%). This huge energy consumption not only increases its operation cost, but also yields negative impact on the environment. One of the most desirable ways is leveraging renewable energy to reduce greenhouse gas emission and amortize data center energy cost (Zhang et al. 2015; Lin et al. 2015).

- Semantic network analysis/Web mining

Semantic network analysis involves areas such as web mining, Natural language processing (NLP) and text analytics. Semantic network analysis is a technique employed to determine a pattern from large network repositories. Semantic network analysis reveals unidentified knowledge on a website and users can use it to perform data analysis. The technique helps to evaluate the effectiveness of a specific website. In order to solve complex tasks in NLP, especially related to semantic analysis, we need formal representation of language i.e. semantic language. NLP is the ability of a computer program to understand human language as it is spoken. NLP is the scientific discipline that is concerned

with making natural language accessible to machines. NLP take up tasks such as extracting relationships from documents, recognizing sentence boundaries in documents, and searching and retrieving of documents. NLP is the way to facilitate text analytics by establishing structure in unstructured text to enable further analysis. Text analytics refers to mining useful information from text sources. It is an umbrella term that describes tasks from annotating text sources with meta-information such as people and places mentioned in the text to a wide range of models about the documents (e.g., text clustering, sentiment analysis, and categorization). In the NLP research, processing huge amounts of textual data has become a major task. As the majority of digital information is present in form of unstructured data such as news articles or web pages, NLP tasks like the cross-document co-reference resolution, event detection or calculating textual similarities frequently need the processing of millions of documents in the stipulated time frame (Agerri et al. 2015; Xia et al. 2016).

Semantic network analysis helps to mine useful information from the web content. The website content consists of audio, video, text, and images. The heterogeneity and lack of structure that permits much of the ever-expanding information sources on the World Wide Web, such as hypertext documents, make the automated discovery, organization, and search and indexing tools of the Internet and the World Wide Web (e.g., Alta Vista, Lycos, WebCrawler, and Meta Crawler) comfortable to users. However, these tools have neither provided structural information nor have they categorized, filtered, or interpreted documents. Therefore, this has prompted researchers to develop more intelligent tools for information retrieval (e.g., intelligent web agents) and extend database and data mining techniques to provide ways of organizing the semi-structured data available on the web. The agent-based approach to semantic network analysis involves the development of sophisticated artificial intelligence systems that can work autonomously or semi-autonomously on the behalf of a particular user to identify and organize web/network information (Castiglione et al. 2017). Some web mining techniques are employed to examine the node and connected structure of a website through graph theory. For instance, in the pattern extraction from hyperlinks within a web-site and analysis of a tree-like structure to describe XML or HTML tags.

- Visualization technique

Visualization techniques are utilized to understand data and interpret them by creating tables, images and diagrams. For instance, the Facebook is using a visualization technique to manipulate and organize the data in its database by intuitive display ways. Big Data visualization is not easy like traditional small data visualization because of the complexity in data. The extension of traditional visualization techniques focus on large-scale data to make data meaningful by using feature extraction and a geometric modeling to reduce the sizes of data before rendering the actual data. For more closely and intuitively data interpretation, many researchers apply batch-model software rendering in a parallel way to obtain the highest data resolution. Data presentation and choosing proper data representation is very important in dealing with big data. Scientists have been realizing that graphical potentialities of the computer and visual strategies for exploiting Big Data would lead to be the data analyst's greatest resource (Fernández-Rodríguez et al. 2017; Gadiraju et al. 2016). As stated by Wang et al. (2016), innovations in data visualization show that a good user interface is worth a thousand petabytes.

- Mathematical/statistical techniques

Statistics is a collection of mathematical techniques that emerge in studies of Big Data to collect, organize and analyze data based on specific fundamental mathematics. Statistical

techniques support decision making in the phase data curation and analysis by exploiting of co-relationships and causal relationships among objectives and the derivation of numerical descriptions of samples. However, traditional statistical techniques are usually not well suited to manage huge volume of data, and some new methods have been developed such as parallel statistics, statistical learning and statistical computing. Especially, scale and parallel implementation of statistical techniques can improve the ability of processing huge volume of data. For example, WalMart stores supports its decision involving advertising campaigns and pricing strategy by exploitation of patterns from transaction data using statistical techniques, associated with machine learning. Batarseh and Latif (2016) proposed statistical regression model for scale linearly with the size of data set and number of processes for estimating functions that are monotonic with respect to input variables.

- Optimization techniques

Optimization techniques have been clarified as efficient techniques to solve quantitative problems in multidisciplinary fields, such as biology, physics, economics and engineering. Different computational strategies such as particle swarm optimization, genetic algorithms, Scheduling algorithm, Bee Colony, evolutionary programming, quantum annealing, and simulated annealing can be efficient for addressing global optimization problems due to their nature of quantitative implementation and because they exhibit parallelism. However, they have high costs in memory and time consumption and it is needed to be scaled up by cooperative co-evolutionary algorithms in a real-time environment to process big data applications. These computational strategies are also married with data reduction and parallelism in optimization problems. Another hot topic of this field is Real-time optimization, whose capability has been demonstrated by decision making problems in many Big Data application such as intelligent transportation systems and wireless sensor networks (Barba-González et al. 2017; Kovalchuk et al. 2018).

6 Classification of research papers

We selected recent articles related to big data from top journals in Science Direct and classified them according to our classification framework. Each article was reviewed and analyzed based on five issues, namely: (1) domain areas and data sources in big data systems proposed in each article (2) datasets used for evaluating the proposed big data technologies (3) the existing big data tools for data processing and storage (4) state-of-the-art techniques used in big data systems. (5) As a result, principles and implications for designing big data systems. The results of our analysis will provide the roadmaps for future research on big data. The details are described below.

6.1 Distribution of articles by publication years and domain areas

The distribution of articles by publication years is shown in Fig. 7. It is clear that publications related to big data steadily increased between 2014 and 2015, and rapidly increased between 2016 and 2017. The decrease of contributions in natural resource management between 2016 and 2017 is thought to be because big data research apparently extended a new application fields in this area such as water and land resources management, energy and utilities management and environmental impact assessment. Based on large volume, and growing data

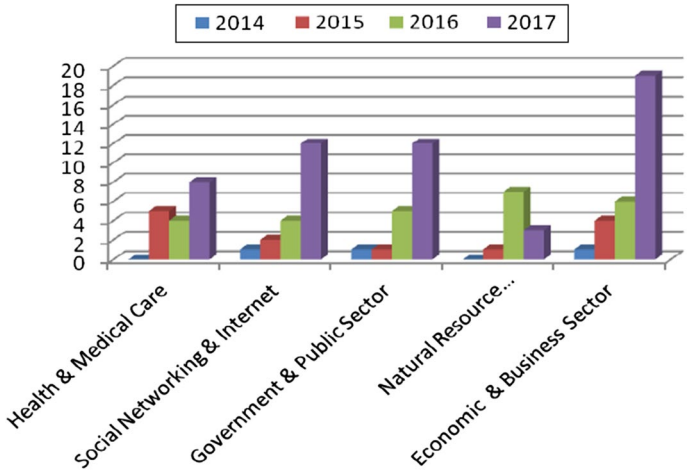

Fig. 7 Distribution of articles by publication year and domain areas

generated by different sources (such as web service, internet and social media), more big-data applications and new application fields are expected to be developed significantly in the future. It shows that interest in big data research will grow significantly in the future.

In general, Big Data has drawn huge attention from researchers in health and medical care, social networking and internet, government and public sector, natural resource management, economic and business sector. Distribution of research papers by domain areas and data sources is shown in Table 1, which shows increases in most of the big data applications during 2017. Most of the existing articles focused on the use of Big Data in economic and business sector (31 out of 93 articles, or 33.3%). It is inferred that although many research related to big data are publishing, few of them were related to natural resource management area (11 out of 93 articles, or 11.8%) and health and medical care (17 out of 93 research papers, or 18.28%) respectively. Therefore, more research looks to be necessary in natural resource management area. We found that the most common sources of data generation were provided from UCI Machine Learning Repository[4] as shown at Table 1. UCI Machine Learning Repository is a collection of free-accessed databases to be used for research purposes on big data. Researchers have conducted experiments by collecting data from UCI Machine Learning Repository to validate their proposed technique in big data.

6.2 Distribution of articles by big data tools

To capture and process the large volume of data and make sense of Big Data, tools (platforms) are needed for example Facebook, Cloudrera and LinkedIn are developing tools for real-time data processing, such as Kafka, and Flume. Twitter is using Storm to gain momentum in real-time data analytics, ETL (Extract, Transform and Load) processing, distributed remote procedure call, online machine learning. In addition, web behemoths like Yahoo, Facebook, Adobe, IBM, eBay are using Hadoop framework for storing and processing large data volumes. Even though the programming languages in these tools

[4] http://archive.ics.uci.edu/ml

Table 1 Distribution of articles by domain areas and data sources

Domain area	Data source	References
Health and medical care	Medical sensors + mobile apps	Kovalchuk et al. (2018)
	Heart Disease Data Set of UCI repository. Checking health status by twitter	Nair et al.(2017)
	Tweets geo using Twitter API	Nguyen et al.(2017)
	Biomedical data (electronic medical records)	Elsebakhi et al.(2015)
	Mobile big data collected from real physical world	Chen et al.(2017)
	Biological meta-database	Pedersen and Bongo (2017)
	Medical sensors	Zhang et al.(2015)
	Microarray database from NCBI GEO[a] repository	Kumar and Rath (2015)
	Sensor data of patients from Cleveland Heart Disease Database (UCI Repository)	Manogaran et al.(2017)
	Historical health data from United Health Foundation[b], MQIC[c], and CMS[d]	Batarseh and Latif (2016)
	Medical records from Respiratory Medicine Department in Lianyungang hospital	Lin et al. (2015)
Social networking and internet	Social media (Twitter data)	Karunaratne et al. (2017)
	Sensor data	Aufaure et al. (2016)
	Social media (Twitter data)	Basanta-Val et al. (2015)
	Social network services (LinkedIn, ResearchGate,Google Scholar) and web technology-related documents	Um et al. (2016)
	Social data (Twitter, online news)	Hidalgo et al.(2017)
	Experimental data (Sina Weibo, a Chinese microblog service)	Ai et al.(2017)
	Multimedia resources (video data)	Guo, J., et al.(2017)
	Media database from Wikipedia, YouTube, Flickr	Guo, K., et al.(2017)
	Web services/browser	Kranjc et al. (2017)
	Web Server log files	Mavridis and Karatza (2017)
	Network service providers	Spivak et al. (2018)
	Sensor Networks	Plimpton and Shead (2014)
	Amazon EC2 Cloud Data sets	Vennila and Kannan (2016)
	Data sets of UCI Repository	Ahmad et al.(2017)
	Tweets data from Twitter	Bharti et al.(2016)

Table 1 (continued)

Domain area	Data source	References
	Multimedia data (Text data sets: Facebook data and Video data sets)	Jayasena et al. (2017)
	Twitter data	He et al. (2015)
	Social (Twitter, Flickr, Panoramio) + Digital Repositories Data (libraries,, multimedia collections) + Wireless Sensor Network + Web Data	Castiglione et al.(2017)
	Social media network (Yahoo Flickr[e])	Amato et al.(2017)
Government and public sector	Datasets from research activities of CityPulse EU project[f] (including Road Traffic Data, Pollution Traffic Data, Parking Data)	De Maio et al. (2017)
	Remote procedure calls	Basanta-Val et al. (2017)
	GPS trajectory dataset[g] + T-Drive trajectory[h]	Wang and Belhassena (2017)
	Wireless sensor network	Mohapatra et al. (2016a, b)
	Netease[i] (Service providers of email, news, microblogging, e-commerce for network security)	Chen et al.(2015)
	PEMS[j] traffic monitoring network to control traffic congestion	Wang et al. (2017a, b)
	Railway network	Oneto et al.(2017)
	Vehicle data (vehicle location, vehicle speed, surrounding vehicles) via mobile application for smart city	Fernández-Rodríguez et al. (2017)
	Web services (CAIDA dataset[k] of Bot attacks)	Singh et al. (2014)
	Twitter data + Smart city data	Kousiouris et al. (2018)
	Sensor data form door failures on metro trains in UK	Manco et al.(2017)
	Sensory data of bridge status (National Bridge Inventory[l] database)	Liang et al.(2016)
	Passing vehicle image (location, time, direction, license plate number)	Xia et al. (2016)
	Transportation and vehicular data (number of vehicles on different roads), parking lot dataset, pollution dataset and toxic gases information	Babar and Arif (2017)
	Residential users' metering data (amount of energy consumed, time of energy consumed to smart city	Jiang et al. (2018)
	Internet of Things (IoT)	Mazhar Rathore et al.(2017)

Table 1 (continued)

Domain area	Data source	References
	IoT dataset from resources such as vehicular network, smart home temperatures, parking place, social media	Mazhar Rathore et al.(2016)
	Streaming data sources managed in Spark: geo-locations and traffic data from city of New York, Twitter	Barba-González et al.(2017)
Natural resource management	Wireless sensor networks	Mohapatra et al. (2016a, b)
	Sensor	Higashino et al. (2016)
	Big data stream computing environments	Sun et al. (2015)
	Experimental data (Grid5000[m])	Hernández et al.(2017)
	Green data centers	Yuan et al.(2017)
	Social networks and information extraction (IE)	Maté et al. (2016)
	Weather service (astronomical data)	Huang et al.(2017)
	Experimental data (Terasort benchmark)	Nghiem and Figueira (2016)
	Electricity datasets	Rahman et al. (2016)
	Datacenter servers (Amazon, Yahoo, Facebook)	Nghiem and Figueira (2016)
	Earth observatory satellite sensors' data	Ahmad et al. (2016)
Economic and business sector	Digital information (textual data calls) from car dataset and wikinews[n] dataset	Agerri et al.(2015)
	Telecom Network service providers	Wang et al. (2017a, b)
	Experimental data (SFinGe large database)	Peralta et al.(2017)
	Experimental data (Higgs, Epsilon, KDD99 dataset)	Eiras-Franco et al.(2016)
	Data from IoT, UCI Repository (Iris, Wine, Shuttle, Online News Popularity, User locations Finland)	Tsai et al. (2017)
	Experimental data UCI and LIBSVM (HIGGS, KDDCup, Poker-Hand, Susy)	Bechini et al. (2016)
	TPC-DS benchmark[o]	Karthik Gadiraju et al. (2016)
	Experimental data from UCI (KDD Cup, Record Linkage Comparison Patterns, Poker Hand dataset)	Del Río et al. (2014)
	Synthetic datasets from size 128 GB to 1 TB	Ding et al.(2017)

Table 1 (continued)

Domain area	Data source	References
	Simulated dataset	Genuer et al.(2017)
	Experimental data (question records, Google Scholar publication records)	Mestre et al.(2017)
	UCI repository (SUSY[p] + HIGGS[q] dataset)	Ghadiri et al. (2016)
	UCI Repository (KDD99)	Chen et al.(2016)
	UCI Repository (c20d10k generated by IBM Generator, chess and mushroom dataset)	Singh et al.(2017)
	Experimental data from multiple temporal events	Ruan and Zhang (2017)
	UCI Repository (Wine data set[r])	Fonseca and Cabral (2017)
	Sales dataset	J.Prajapati et al.(2017)
	Synthetic dataset from a live online test application	Singh and Bawa (2017)
	UCI Repository (Kddcup, Poker Hand, Susy, RLCP)	Triguero et al. (2015)
	UCI repository (Higgs, Susy)	Elkano et al. (2017)
	MNIST handwritten digits dataset	Zhang et al. (2016)
	UCI repository (Mushroom)	Qian et al.(2015)
	Experimental data including cardholders and credit card transactions	Carcillo et al. (2018)
	UCI- Human Activity Recognition dataset Using Smartphones Data Set	Tennant et al.(2017)
	TPC-H benchmark[s]	Sahal et al.(2017)
	Experimental data UCI (RLCP, Census, Fars, Shuttle)	Pulgar-Rubio et al.(2017)
	Experimental data UCI (Susy, PokerHand, Higgs)	Maillo et al.(2017)
Health and medical care + Government and public sector	PEMS-SF[t] dataset and Kent Ridge Breast Cancer[u]	Apiletti et al.(2017)
	image data (cancer blood cells, MRI scan, satellite image on drought prone lake and region)	Tripathy and Mittal (2016)
Health and medical care + Economic and business sector	Experimental data UCI (Covertype, eCO, Higgs, Kddcup, PokerHand, Susy)	Ferranti et al.(2017)
	Experimental data UCI (Epsilon, Splice)	Arias et al.(2017)
	Real datasets (MNIST[v], DNA and KDDcup99[w])	Huang et al.(2016)
	Experimental data (Flower[x], Covtype, Poker, Shuttle, Breast Cancer,, Pendigits[y])	Wang et al. (2015)

Table 1 (continued)

[a]GEO, http://www.ncbi.nlm.nih.gov/gds/

[b]http://www.americashealthrankings.org/

[c]http://mqic.org/

[d]http://www.americashealthrankings.org/

[e](YFCC100 M) https://webscope.sandbox.yahoo.com

[f]http://www.ict-citypulse.eu/page/

[g]https://www.microsoft.com/en-us/research/publication/geolife-gps-trajectory-dataset-user-guide/

[h]http://research.microsoft.com/apps/pubs/?id=152883

[i]http://www.163.com

[j]PeMS project, https://pems.eecs.berkeley.edu/

[k]https://data.caida.org/datasets/security/telescope-3days-conficker/

[l]https://www.fhwa.dot.gov/bridge/

[m]https://www.grid5000.fr

[n]http://en.wikinews.org

[o]DSGen v1.1.0

[p]http://archive.ics.uci.edu/ml/datasets/SUSY

[q]http://archive.ics.uci.edu/ml/datasets/HIGGS

[r]http://archive.ics.uci.edu/ml/, 2013

[s]http://www.tcp.org/hspec.htm

[t]https://archive.ics.uci.edu/ml/datasets/PEMS-SF

[u]http://mldata.org/repository/data/viewslug/breast-cancer-kent-ridge-2/

[v]http://yann.lecun.com/exdb/mnist/

[w]http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[x]http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

[y]http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

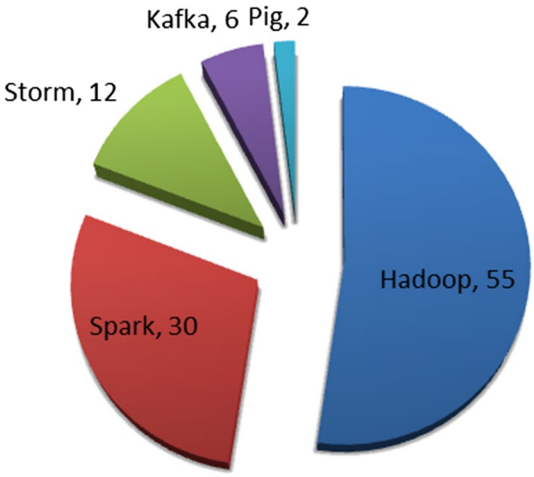

Fig. 8 Distribution of papers by the most popular big data tools

are different, they all provide highly efficient and scalable structure to store and process Big Data workloads (Carcillo et al. 2018; Wang et al. 2017a, b). Distribution of articles by big data tools is represented in Fig. 8.

Among big data platforms, Hadoop have been used the most often in different application fields (55 out of 93 articles, or 59.1%). Because, Hadoop is first- established framework based on Map-Reduce programming model that is very effective for large scale structured, unstructured and semi-structured data, for information processing and retrieval. Also, Hadoop has built-in features, such as fault tolerance, high scalability, and adaptation to heterogeneous clusters.

Apache Spark are gaining great popularity as next generation replacement for Hadoop framework as thirty articles (30 out of 93 articles, or 32.26%) made use of Apache Spark for massively parallel data analytics. Because, Spark is an attractive platform for big data processing applications to hide most of the complexities related to fault tolerance, parallelism, and cluster setting from application developers. In particular, the spark improves Hadoop MapReduce in terms of flexibility and performance in the programming model, especially for iterative applications. It can support both batch applications and streaming applications in real-life data science, while providing interfaces to other established big data technologies, such as HDFS and NoSQL databases for storage. It is important to mention, despite the fact that Spark has been designed to manage iterative applications; it also has been designed to handle MapReduce-based workflows. Thus, MapReduce-based workflows can be improved (by usage of RDDs) into Spark framework.

Storm is best known platform for data stream processing that has been used by recent contributions (12 out of 93 articles, or 12.9%). It is inferred that although many articles were published in development of big data applications, few of them made the usage of Kafka (6 out of 93 articles, or 6.45%), and Pig, S4, Flume, Flink. Because, these big data platforms are still new and they are emerging to tackle the problem of big data in recent contributions. Therefore, it looks to be emerged more in future research papers. The following Table 2 shows a comparison between big data tools based on its advantages and disadvantages.

Table 2 Big data tools based on batch and stream processing

Big data tools	Advantages	Disadvantages
Hadoop	_ Integrates, transforms, stores and classifies Petabytes of data with high fault-tolerance and scalability but not fast performance _ The code development is simple since the developers do not have to deal with complexities of MapReduce coding _ HBase as Hadoop storage accesses large-scale heterogeneous real-time or historical data _ Hive provides a data warehouse infrastructure to provide data summarization, ad hoc querying and analysis	_ Relies on hard drives that repeatedly read or save data in the disk, this affects execution time and makes it unsuitable for real-time applications _ Processing Petabytes of data from minutes to weeks _ Require casting any computation as a MapReduce job
Spark	_ A cluster computing system to make faster real-time data analysis _ Load data once and keeps it in memory for iterative-computational processes _ Enables distributed and parallel computations and makes computations practically and efficiently by performing in-memory computations _ Hide complexities related to fault-tolerance, parallelism, and cluster setting from developers _Suitable for SQL queries, streaming data, graph processing, machine learning (MLlib) _ Power on existing Hadoop clusters and seamlessly can coordinate with HDFS and HBase for data access _ Optimize I/O access, higher performance (up to one hundred times) and more flexible in comparison with Hadoop by exploitation of main memory instead of the disk	_ Reduce the computational times but with heavier computational costs than Storm _ Develop a stable and efficient machine learning algorithm on Spark is difficult and not straight forward
Storm	_ Fault-tolerant, parallel and distributed real-time computation system to process unbounded streams of data very fast _ use graphical user interface easily _ Is scalable, simple to set up and operate, fault-tolerant, can be used with any programming language _ Guarantees the processing of data, prevent message loss and support horizontal scalability	_ Do not explicitly assign different parts of the application to different physical nodes, which is typically required in real-time applications _ Lack the notion of scheduling parameters enforced in different computational nodes
S4	_ An effective framework alternative for unbounded streams of real-time data _ Proven, fault-tolerant, scalable, distributed, pluggable platform _ Use graphical user interface easily	_Can potentially lead to message loss due to its recovery mechanism

Table 2 (continued)

Big data tools	Advantages	Disadvantages
Kafka	_ Low latency, high throughput message handling and immutable activity data _ Support very long and fast stream of data due to fast communication and distribution by using low latency techniques _ Realize message brokers and the publish–subscribe communication	_ Require a full set of co-ordination and management in messaging system _ In delivering messages to the consumer, the broker in Kafka employs certain system calls, but if the messages need some tweaking, this decreases the performance of Kafka
Pig	_ Support MapReduce-enabled query on huge data sets _ Provide more functionality by extending with User Defined Functions (UDF) _ Has more capabilities based on performance in comparison with Hive and is more suitable for regularly scheduled work	_ Scheduling MapReduce jobs in Pig takes time _ Complex applications require many UDFs so loses its simplicity
Flume	_ A distrusted data collection service that collect, save, and move a large amount of streaming data (i.e., the sensory data about bridge status) into Hadoop from a variety of sources	_ Do not guarantee that message reaching is not unique (duplicate messages might pop in at times, in many scenarios)
Flink	_ Support batch, stream processing, iterative processing, interactive processing and graph processing in-memory computations with high-performance and low latency dataflow architecture	_ Cannot support explicit data catching
Sqoop	_ Support the ingestion of log data, which is related to bridge design and operation such as transportation status, bridge configuration (e.g., National Bridge Inventory), and weather conditions	_ Slow in complex queries because it uses MapReduce jobs _ An atomic step as cannot be paused and resumed (failure recovery problem). If failed, it is needed to clear things up and start again
YARN	_ A cluster resource manager to help in decoupling the programming platform from the resource management in cluster _ Control memory resources and the CPU in the nodes in cluster	_ Do not involve problem of optimizing application and cluster performance
Cassandra	_ A distributed database designed for scalability that support replication across multiple nodes or data centers _ Support fault tolerance, low latency, and linear scalability when querying and manages consistency of requests at the node level _ Avoid a Node failure happen because data is stored on multiple nodes organized in a ring shape (there is no master and every node is as important as the others)	_ Require the setting of some parameters (primary key) having an impact on performances for creation of a Cassandra table

Table 3 Distribution of articles by big data tools and techniques used

Big data tools based on batch processing	Big data techniques with their efficiency for related domain	References
MapReduce programming model on Hadoop	ML[a] (Random Forest Tree, Tree C4.5) for energy consumption predictions	Maté et al. (2016)
	ML (Graph algorithms) on sensor networks	Plimpton and Shead (2014)
	ML (Apriori + FP-Growth algorithm) for economic and business sector	Bechini et al. (2016)
	ML (Random Forest classifier) for economic and business sector	Del Río et al. (2014)
	ML (Random forest) for economic and business sector	Genuer et al.(2017)
	ML (Neighborhood Method) for economic and business sector	Mestre et al.(2017)
	ML(Fuzzy C-Means clustering) for economic and business sector	Ghadiri et al. (2016)
	ML (Neighborhood Method) for economic and business sector	Chen et al.(2016)
	ML (Association Rule Mining) for economic and business sector	Singh et al.(2017)
	ML (Sequential learning) on health and economic data	Huang et al.(2016)
	ML (support vector machine) on health and economic data	Kumar and Rath (2015)
	ML (Artificial Neural Network) for economic and business sector + natural resource management	Fonseca and Cabral (2017), Rahman et al. (2016)
	ML (item-set mining algorithm) for health and government sector	Apiletti et al.(2017)
	ML (Pattern mining algorithm) for economic and business sector	J.Prajapati et al.(2017)
	ML (B-tree) for economic and business sector	Singh and Bawa (2017)
	ML (Prototype reduction technique) for economic and business sector	Triguero et al. (2015)
	ML (Sequential learning) for health and economic sector	Wang et al. (2015)
	ML (Fuzzy Rule-Based Classification) for economic and business sector	Elkano et al. (2017)

Table 3 (continued)

Big data tools based on batch processing	Big data techniques with their efficiency for related domain	References
	ML (C-means clustering, fuzzy clustering) for health and government sector	Tripathy and Mittal (2016)
	ML (Deep learning) for economic and business sector	Zhang et al. (2016)
	ML (Bayesian network) for economic and business sector	Liang et al.(2016)
	ML (Fuzzy clustering) for economic and business sector	Ding et al.(2017)
	ML (Random Forests) for government and public sector	Singh et al. (2014)
	ML (Pattern recognition) + Mathematical/Statistical techniques for government and public sector	Rathore et al.(2016)
	Cloud computing + ML (Naive Bayes classifier) for social networking and internet	Kranjc et al. (2017)
	Cloud computing + ML (Bayes Classifier) for social networking and internet	Vennila and Kannan (2016)
	Cloud computing for healthcare applications	Zhang et al.(2015), Lin et al. (2015)
	Mathematical techniques (Parallel algorithm) for economic and business sector	Qian et al. (2015)
	Optimization techniques (Ant colony) for Social networking and internet	Jayasena et al. (2017)
	Optimization technologies for natural resource management and government and public sector	Nghiem and Figueira (2016), Babar and Arif (2017), Pedersen and Bongo (2017)
	Optimization technique (Genetic algorithm) for social networking and internet	Spivak et al. (2018)
	Optimization technique (Bee Colony) for social networking and internet	Ahmad et al.(2017)
	Optimization technique (privacy-preserving query) for economic and business sector	Jiang et al. (2018)
	Optimization technique (Fusion algorithm) for natural resource management	Ahmad et al. (2016)
	Visualization technique for economic and business sector	Ruan and Zhang (2017), Karthik Gadiraju et al. (2016)
	Statistical analysis (Linear regression) for health and medical care	Batarseh and Latif (2016)

Table 3 (continued)

Big data tools based on batch processing	Big data techniques with their efficiency for related domain	References
Hadoop (HBase,Hive) + Pig	Semantic network analysis (Text analysis) + ML (N-Gram model) for social networking and internet	He et al. (2015)
	Semantic network analysis for government and public sector	Xia et al. (2016)
	Mathematical/Statistical techniques for government and public sector	Manco et al.(2017)
	ML (Stochastic gradient descent algorithm + logistic regression) for health and medical care	Manogaran et al.(2017)
Big data tools based on stream processing	**Big data techniques with their efficiency for related domain**	**References**
Spark	Visualization technique for government and public sector + natural resource management	Kousiouris et al. (2018), Kumar Mohapatra et al. (2016a, b)
	ML (SVM, Decision Tree, Logistic Regression, k-Means) for natural resource management	Hernández et al.(2017)
	ML (Random-tree) for government and public sector	Wang and Belhassena (2017)
	ML (Fuzzy rule-based classifiers) for health and economic sector	Ferranti et al.(2017)
	ML (Decision tree algorithm) for health and medical care	Nair et al.(2017)
	ML (K-Nearest Neighbors) for economic and business sector	Maillo et al.(2017)
	ML (Bayesian Network Classifiers) + Cloud Computing for health and economic sector	Arias et al.(2017)
	ML (Kernel function) for health and medical care	Nguyen et al.(2017)
	ML (Feature selection methods) for economic and business sector	Eiras-Franco et al.(2016)
	ML (K-Means clustering + Naive Bayes) for government and public sector	Chen et al.(2015)
	ML (Deep learning)	Oneto et al.(2017)

Table 3 (continued)

Big data tools based on stream processing	Big data techniques with their efficiency for related domain	References
	ML (k-means clustering) + Optimization techniques (Genetic, Particle swarm optimization) for economic and business sector	Tsai et al. (2017)
	Mathematical technique for natural resource management	Nghiem and Figueira (2016)
	Semantic network analysis for social networking and internet + health and medical care	Guo et al.(2017), Chen et al.(2017)
	Semantic network analysis + ML(Clustering algorithm) for social networking and internet	Ai et al.(2017)
	Cloud computing + Visualization technique for natural resource management	Mohapatra et al. (2016a, b)
	Cloud computing for social networking and internet	Guo et al.(2017)
	Cloud computing for natural resource management	Yuan et al.(2017)
	Optimization technique for government and public sector	Barba-González et al.(2017)
Storm	Cloud computing for natural resource management	A. Higashino et al. (2016)
	Optimization techniques for government and public sector	Basanta-Val et al. (2015), Sun et al. (2015), Kovalchuk et al. (2018)
	Semantic network analysis for government and public sector + social networking and internet	Basanta-Val et al. (2017), Um et al. (2016), Agerri et al. (2015)
	ML (Frequent pattern mining) for telecom network	Wang et al. (2017a, b)
	ML (clustering algorithm) for social networking and internet	Karunaratne et al. (2017)
	ML (Fuzzy sets) for government and public sector	De Maio et al. (2017)
Spark + Kafka (as event bus)	ML (Deep reinforcement learning) for traffic monitoring network	Wang et al. (2017a, b)
	ML (Random Forest classifier) for economic and business sector	Carcillo et al. (2018)
Kafka	Visualization technique for government and public sector	Fernández-Rodríguez et al. (2017)
Kafka + Storm + Zookeper as coordinator	Semantic network analysis and cloud computing for social networking and internet	Castiglione et al. (2017)

Table 3 (continued)

Big data tools based on stream processing	Big data techniques with their efficiency for related domain	References
S4	ML (Graph-based) for social networking and internet	Hidalgo et al.(2017)
RDF Streams	Semantic network analysis for social networking and internet	Aufaure et al. (2016)
Big data tools based on hybrid processing	**Big data techniques**	**References**
Hadoop + Spark	Cloud computing on astronomical data	Huang et al.(2017), Mavridis and Karatza (2017)
	Artificial neural networks for economic and business sector	Peralta et al.(2017)
	ML (Fuzzy sets) for economic and business sector	Pulgar-Rubio et al.(2017)
	ML (Pattern recognition + Decision models) for government and public sector	Rathore et al.(2017)
	ML (Content based + Collaborative filtering + Co-clustering) for social networking and internet	Amato et al.(2017)
	ML (New functional networks classifier) for health and medical care	Elsebakhi et al.(2015)
Hadoop (Hive) + Flume	Semantic network analysis for social networking and internet	Bharti et al.(2016)
Hadoop + Kafka	ML (Nearest Neighbour) for economic and business sector	Tennant et al.(2017)
Hadoop (Hive) + Flink	Optimization technique for economic and business sector	Sahal et al.(2017)

[a]ML is abbreviation Machine learning and data mining algorithms

6.3 Distribution of articles by big data tools and techniques

Distribution of articles by big data tools and techniques used in data analytics is represented in Table 3. In recent years the success of big data platforms such as Spark, Hadoop, Flink and etc. was supported by the advent of open source machine learning and data mining libraries. First attempt that support a collection of scalable Machine Learning algorithms for Big Data is Mahout for Apache Hadoop. Mahout provides distributed implementations of well-known classification, clustering, and item-set mining algorithms based on MapReduce. MLLib is the data mining and Machine Learning library born in 2012 as an extra component of Spark to be released and open-sourced in 2013. Researchers exploit Spark features by MLlib to implement faster iterative procedures. Finally, FlinkML is an attempt of scalable machine learning algorithms and an intuitive API to users that is built on Flink, and provides some machine learning algorithms including SVM, k-NN for unsupervised learning, and Multiple Linear regression for supervised learning. As shown at Table 3, the majority of the researchers employed machine learning and data mining algorithms (53 out of 93 research papers, or 57%) to efficiently analyze large amounts of data in big data systems. Because popular parallel data processing platform such as Spark, Hadoop and Flink are especially suitable for scalable machine learning and data mining algorithms. Researchers have also conducted studies on optimization techniques (14 out of 93 research papers, or 15%) to solve quantitative problems in big data applications. Because, they are useful for addressing global optimization problems and can scale up the large-scale optimization by cooperative co-evolutionary algorithms. In addition, due to the impressive features of cloud computing such as, pervasive service-oriented nature and elastic computing power, numerous efforts have been made to develop cloud computing technologies (13% of papers) in big data area especially in healthcare applications for mining large-scale medical data. Currently, only a few research papers were adopted by applications of web mining, visualization techniques and mathematical and statistical techniques on big data analysis purposes as shown at Table 3.

7 Conclusion and future work

We conclude that big data can play an important role in the coming technological leapfrogging and for decision-making purposes. Hence, big data have attracted huge attention of academics and practitioners. In this research, we have identified 93 articles on big data and classify them based on our classification framework, to understand the trend of big data-related research and to provide practitioners and researchers with guideline for future research on big data. To make the best use of the relevant articles returned by the search engines, we have kept the search string not too specific but still reflecting what we have wanted to search for. Moreover, the search string has been used for searching not only for the titles, abstracts but also for the full text. We conduct search on articles that were selected solely from top academic journals not conferences and books. However, using clearly predefined inclusion/exclusion criteria, papers have been selected, and then reviewed. From these primary big data papers, we have extracted and synthesized the data to give an overview on current frames of big data technologies for domain areas including health and medical care, social networking and internet, government and public sector, natural resource management, economic and business sector. In addition, we provided an overview of requirements and technologies considered for current big data platforms such

as sources of data generation, big data tools for data processing and storage of data and big data analytics techniques. The results of this paper were based on the data extracted and synthesized from the selected big data studies. Based on previous publication rates, interest in big data related research will grow significantly in the future. Based on our experiences, we feel that big data technology will continue to evolve together with bigger size data and better user requirement. On the other hand, from our experiences dealing with clients and application developers, we discover that the following ideas and technologies are important in making breakthroughs in big data analytics and bringing its application to another level:

Firstly, the main function of big data system is to manage different types of data that come from different sources. To illustrate, Netease maintains data from email system, news system, microblog system and online e-commerce transaction system for security reasons. A smart city system manages data from mobile phones, sensors, cameras, and many other devices so that general security, public safety and effective traffic control can be managed and monitored. Health and medical care systems obtain sensor data from patients in offering them prognostic interventions, novel therapies, as well as shaping their lifestyle and behaviors. Road traffic monitoring systems maintain data from surveillance cameras and analyze the large amount (hours/days) of video footage in order to catch traffic violators, detect road accidents, track criminals, collect evidences for investigation. It is taxing to scrutinize big data; since these data is of different formats. To do the scrutinizing of the data, it needs a relational table, a social graph and a text-format log file. There is no big data tools which are available that can perform the diagnostic job among the hybrid data formats. GraphX[5] is recently released to support relational data and graph data. epiC can also handle analyzing multi-format data. Unfortunately, they are only limited to specific types of data in an explicit way (users have to write the codes for supporting different data formats by themselves). To develop big data applications on hybrid data formats, we should formally identify a set of operators to process the mixture of data and translate them into efficient jobs on top of the existing processing engines like Hadoop, Spark.

Secondly, in developing the characteristics of a new hardware, the existing distributed algorithms should be restored. The big data algorithm must be fully optimized for specific configuration of hardware. Although a number of algorithms have been suggested for new hardware and hybrid framework, they are mainly targeted at the single-node system. A future cluster node may be endowed with hybrid hardware consisting of CPU, SSD, DRAM, GPU, and HDD. In handling the big data applications, the scalable distributed version is needed.

Thirdly, big data system is still mysterious to many end users. Although the analytic tools are provided, they are unaware still of the system's capabilities. Hence, a user-friendly visualization technique is necessary to narrow this gap between big data system and its users. The visualization technique should show the analytic results in users' perceptive so that they can effectively identify the interesting results.

Lastly, big data applications are still limited in areas like e-commerce, biology and finance. Since big data analytics is still in its infancy stage of development, the techniques and tools are then inadequate to fully solve the real big data problems. In fact, some of these techniques and tools cannot even be viewed as big data tools in the true sense. Due to the fast development in big data applications, it is projected that new applications will be materialized when we combine big data technology with other conventional industries. In

[5] https://spark.apache.org/graphx/.

doing this, these applications will pose new requirements for big data systems, thus driving us to look for and suggest new solutions. However, it can be said that big data research in a smart city is still in its infancy and the solution to the discussed challenges has proven its practicality. E-businesses and organizations could also be assisted through needs analysis tools, which contribute in the delivery of products and services which meet their customers' requirements. The growing modernized urban areas and increasingly connected rural areas require the development of efficient transportation infrastructure to fill the needs of visitors and commuters. The dynamic wireless communication techniques, computational intelligence and data analytics capabilities, mobile cloud computing, and context-aware technologies are able to increase more research works and commercial applications in the transportation area. The volatile growth in the number of devices connected to the Internet of Things (IoT) and the exponential increase in data consumption reflect the perfect overlapping of big data with that of IoT. The management of big data in a progressive and expanding network has increased concerns about the efficiency of data collection, data processing, data analytics, and data security. To address these concerns, researchers need to examine the challenges which are related to the successful deployment of IoT. The union of big data analytics and IoT has created several opportunities for developing more successful applications and decreasing the problems or concerns that occur.

Many quarters are optimistic about the development and the adoption of big data technology. It is expected that more users will update their systems using big data technology, as they will meet more success stories on big data applications. Therefore, more and huge investment from both government and private sectors should be coming in to capture more gains from big data. Big data application is significant since we need more superior storage and I/O techniques, more beneficial computer architectures, more efficient data-intensive techniques such as biological computing, social computing, and cloud computing, and more advanced technology in handling big data in other areas with different platforms but with sound architecture, approach, infrastructure, and properties. We are fortunate to witness the birth and the development of Big Data, and in managing big data, no man is an island. All involved parties like human resources management, capital investment management, innovative and creative catalysts are fundamental in the development of big data application.

Acknowledgements This work is supported under the university Research Entity Initiatives Grant (600-RMI/DANA 5/3/REI (16/2015)). We thank IRMI (Institute of Research, Management and Innovation), UiTM for their continuous support.

Appendix: recent contributions on big data

See Table 4.

Table 4 Focus of recent contributions on big data

Focus	References
Present a cloud based platform for big data with perfect linear speedup and stream processing to import web services as workflow components	Kranjc et al. (2017)
Present a big data infrastructure to deal with digital contents in cultural heritage environments such as multimedia contents, social data. This paper query, analyze and process digital cultural contents from heterogeneous and distributed sources and present needs of users in a suitable format	Castiglione et al. (2017)
Present architecture to integrate internal, external, structured, unstructured data to improve energy consumption predictions	Maté et al. (2016)
Present a distributed associative classification according to the MapReduce programming to manage big data	Bechini et al. (2016)
Present a big data architecture to analyze the data generated from the camera sensors to form the barrier and to detect behavior of the intruders by a cloud layer designed	Kumar Mohapatra et al. (2016a, b)
Present a scalable intrusion detection system for Botnet attacks to handle heavy network bandwidths	Singh et al. (2014)
Present a simulator of complex Event Processing systems in cloud environments	A. Higashino et al. (2016)
Present a Fuzzy Concept Analysis on a distributed real-time computation system for stream processing in smart cities	De Maio et al. (2017)
Present a distributed system based on stream clustering algorithm for fast data	Karunaratne et al. (2017)
Present the real-time analytics integrated with semantic technologies over big data	Aufaure et al. (2016)
Improve the predictability of big-data application by combining stream processing technology and real-time	Basanta-Val et al. (2015)
Dealing with imbalanced big data using random forest classifier	Del Río et al. (2014)
Present a predictable model for remote procedure calls running an cluster of stream processors	Basanta-Val et al. (2017)
Present an energy-efficient resource scheduling to achieve low response time and high energy efficiency in big data stream computing environments	Sun et al. (2015)
Present a semantic complex event processing model for analyzing research activities from several information sources	Um et al. (2016)
Present an integrated framework including Analytics services, Smart City platform, and Twitter for social network data to identify large Crowd Concentration events with affecting on user journey	Kousiouris et al. (2018)
Present a task-level adaptive MapReduce framework to process streaming data in healthcare applications	Zhang et al.(2015)
Present an asteroid system on distributed cloud environment to help astronomers with analyzing large astronomical data to discover asteroids	Huang et al.(2017)
Present architecture for clinical decision support systems in ambulance control to integrate various data sources including stored and streaming data of diverse formats and nature	Kovalchuk et al. (2018)
Present an alarm behavior analysis for telecom networks to establish parent–child rules to show operation patterns from a large number of alarms	Wang et al. (2017a, b)
Present the performance of log file analysis (such as Facebook logs) in cloud computational frameworks	Mavridis and Karatza (2017)

Table 4 (continued)

Focus	References
Present a parallel real-time data stream classifier for data stream mining	Tennant et al.(2017)
Present a processing graph system to be adapted to dynamism of the flow of events by detecting peaks of traffics and predicting the traffic using past behavior of the flow of events	Hidalgo et al.(2017)
Present an algorithm for categorizing of data in Map-Reduce frameworks and optimizing file location within storage (HDD, RAM, SDD)	Spivak et al. (2018)
Present a framework for Streaming data analytics in a distributed-memory manner	Plimpton and Shead (2014)
Present a job profiling technique for optimal resource provisioning for MapReduce workload	Nghiem and Figueira (2016)
Present a distributed and parallel processing of massive textual data in form of unstructured data by Natural Language Processing for calculating textual similarities	Agerri et al.(2015)
Present a system for urban planning and smart cities using an IoT-generated Big Data analysis	Rathore et al.(2016)
Present an method to configure the parameters of Spark to effect the performance of Spark workloads	Bei et al.(2018)
Present a machine learning model to recommend optimal task configuration for task parallelization to predict duration of Big Data workloads	Hernández et al.(2017)
Present a R-tree index in distributed system for processing trajectory search	Wang and Belhassena (2017)
Present a decomposition methodology for fingerprint recognition in Big Data frameworks in which matching process is decomposed into smaller steps	Peralta et al.(2017)
Present a Multi-objective evolutionary based fuzzy system in big data to maximize accuracy and minimize complexity	Ferranti et al.(2017)
Present a health status prediction system on streaming big data using application of Spark based machine learning	Nair et al.(2017)
Present an evolutionary fuzzy algorithm to discover subgroups in big data environments	Pulgar-Rubio et al.(2017)
Present big data analytic architecture to analyze the data generated during the barrier construction and detect intruder in wireless sensor networks using camera sensors	Mohapatra et al. (2016a, b)
Present Kernel-based features for population health prediction from social media data	Nguyen et al.(2017)
Present a job profiling method for performance efficiency of task resource provisioning in MapReduce workload	Nghiem and Figueira (2016)
Present an iterative MapReduce process based on spark for the k-Nearest Neighbors on big data	Maillo et al.(2017)
Present distributed version of Bayesian Network Classifiers for big data under MapReduce with Spark	Arias et al.(2017)
Present a data filling algorithm on Spark framework to complete the incomplete energy big data issue in green data centers	Yuan et al.(2017)
Present a Real-time Fraud Finder to integrates Big Data tools with a machine learning approach	Carcillo et al. (2018)
Present a dynamic Big Data Optimization technique on Spark clusters to minimize distance and travel time in transportation	Barba-González et al.(2017)

Table 4 (continued)

Focus	References
Present an Hot Topic Detection Method from large number of textual digital materials for Microblogs on Spark	Ai et al.(2017)
Present a predictive modeling classifier to predict cancer specific outcomes with dealing on imbalanced and sparse clinical real-life data	Elsebakhi et al.(2015)
Present new implementation of feature selection algorithms based on multithreaded processing and distributed version of Spark to speed up feature selection	Eiras-Franco et al.(2016)
Present a analytic big data system for real-world application to handle jobs such as user pattern mining, email spam detection, game log analysis	Chen et al.(2015)
Present an object detection method under mobile distributed computing architecture to handle Multimedia big data	Guo et al.(2017)
Present Symmetric Matrix-based Bayes Classifier for computation and Big Data processing in Cloud environment	Vennila and Kannan (2016)
Present a prediction method for making decision online based on deep reinforcement learning	Wang et al. (2017a, b)
Present a big data storage model to cluster huge dimensions of big data, and gains the non-key and key dimension clusters	Ding et al.(2017)
Present a big data fusion scheme for heterogeneous media data to extract semantic information in internet of things environments	Guo et al.(2017)
Present a Random Forests with focus on classification problems to deal with big data in offline or online contexts	Genuer et al.(2017)
Present a train delay prediction system for large amount railway networks to use historical train movements data to manage traffic and dispatching processes	Oneto et al.(2017)
Present approach for managing the large-scale biological data to provide efficient storage and runtime generation of meta-database versions	Pedersen and Bongo (2017)
Present a multimedia recommender system to recommend multimedia objects of user's interest according to her/his preferences in online Social networks	Amato et al.(2017)
Present an automatic mechanism of data partitioning using Sorted Neighborhood Method to perform load balanced (parallel entity matching)	Mestre et al.(2017)
Present an architecture to select features and process Big Data using Artificial Bee Colony based social Internet of Things	Ahmad et al.(2017)
Present a parallel real-time data stream classifier to address overlap of the Velocity and Volume aspects of Big Data	Tennant et al.(2017)
Present a method designed on Hadoop platform to analyze entire collection of records with an optimized map and reduce functions to leading higher performance and scalability	Ghadiri et al. (2016)
Present a Smart City security system with intrusion detection and communication security protocol for communication between remote smart system/User and city building	Rathore et al.(2017)
Present a prediction algorithm to characterize disease dynamics by analyzing data collected by the mobile devices with impact of network structure on disease dynamics during an epidemic	Chen et al.(2017)
Present parallel attribute reduction in neighborhood dominance relation matrix with focus among categorical and numerical attribute values	Chen et al.(2016)

Table 4 (continued)

Focus	References
Present algorithm to tackle the performance optimization of Apriori algorithm in managing big data based on Hadoop cluster and reducing the number of passes	Singh et al.(2017)
Present parallel ensemble of sequential extreme learning based on MapReduce to analyze massive data efficiently and accurately	Huang et al.(2016)
Present support vector machine classifier on MapReduce framework for selecting relevant genes in microarray data and classification of data	Kumar and Rath (2015)
Present a system for optimizing query processing in Big Data to reduce cost of data processing (i.e., write, read) using the coarse granularity of reused-based opportunities of results loaded from slow storage	Sahal et al.(2017)
Present algorithm for extracting timings and durations of sequential patterns from big dataset to analyze multi-modal data streams by integrating scalable computing, information visualization, and user interfaces	Ruan and Zhang (2017)
Present a Neural Network for extracting information from big data via deep-Learning and graphical processing units	Fonseca and Cabral (2017)
Present MapReduce architecture to parallelize and speed up the item-set extraction and frequent item-set mining from high-dimensional datasets	Apiletti et al.(2017)
Present MapReduce architecture to overcome the limitations of association rule mining in processing speed for analyzing big data which do processing of transactional data into clusters and transfers task to each nodes in a cluster	J.Prajapati et al.(2017)
Present Hadoop framework to detect and analyze the sarcasm sentiment in social media text using natural language processing	Bharti et al.(2016)
Present a bridge health monitoring system based on bridge data such as weather conditions, traffic status, and bridge structural to evaluate the serviceability analysis for bridges	Liang et al.(2016)
Present an application to predict door failures on metro trains through fault detection from diagnostic data and big data analysis on sensor streams	Manco et al.(2017)
Present heart disease prediction model with implementation of IoT to collect and process sensor data (big data) generated from body sensor devices for secured smart healthcare monitoring and predicting the heart diseases	Manogaran et al.(2017)
Present big data analytics to healthcare data from multiple sources to assess quality insights of the field	Batarseh and Latif (2016)
Present a parallel B-Tree index to optimize data access and search query execution time on MapReduce	Singh and Bawa (2017)
Present home-diagnosis service in cloud-based framework to get diagnosis based on similar patients' records	Lin et al. (2015)
Present a recommendation service for a smart city to gather and analyze information from drivers in a city to improve driving efficiency and safety on roads	Fernández-Rodríguez et al. (2017)
Present a distributed partitioning solution for prototype reduction techniques to enhance nearest neighbor classification when dealing with big data	Triguero et al. (2015)
Present data clustering algorithms based on parallel computing on a cloud computing environment to reduce running time of algorithms for data clustering	Tsai et al. (2017)

Table 4 (continued)

Focus	References
Present a parallel technique to improve performance of online sequential learning for large scale data by parallelizing hidden layer output of matrix calculations in sequential learning algorithm	Wang et al. (2015)
Present architecture for multimedia big data analyzing on cloud platform that reduces time for data distribution and transcoding data into specific formats based on user requirements.	Jayasena et al. (2017)
Present a Fuzzy Rule-Based Classification on imbalanced datasets to tackle classification accuracy	Elkano et al. (2017)
Present a power generation prediction system to predict the amount of power required to electricity consumption for United States	Rahman et al. (2016)
Present a possibilistic approach to clustering algorithm based on MapReduce paradigm for image segmentation in form of cancer blood cells, brain MRI and satellite images.	Tripathy and Mittal (2016)
Present framework for traffic data processing to analyze data in intelligent monitoring and recording system for monitoring traffic in cities and vehicle trajectory tracking	Xia et al. (2016)
Present distributed learning algorithm to reduce time of learning processes in deep learning algorithms and scale up Big Data learning	Zhang et al. (2016)
Present a smart city architecture based on Big Data analytics to provide smart parking service to citizens, selecting empty spaces in parking lots and water consumption	Babar and Arif (2017)
Present a similarity query approach over multidimensional metering data in in encrypted form in a distributed system to protect customers' privacy in smart grids	Jiang et al. (2018)
Present divide-and-conquer mechanism in machine-to-machine communication that does not require whole set of data to be processed and to kept in main memory	Ahmad et al. (2016)
Present hierarchical attribute reduction algorithm using MapReduce and task parallel for big data	Qian et al. (2015)
Present a social analytics framework with sentiment benchmarks to enhance marketing intelligence by customer opinions from social media	He et al. (2015)

References

Agerri R, Artola X, Beloki Z, Rigau G, Soroa A (2015) Big data for natural language processing: a streaming approach. Knowl-Based Syst 79:36–42

Ahmad A, Paul A, Rathore MM (2016) An efficient divide-and-conquer approach for big data analytics in machine-to-machine communication. Neurocomputing 174:439–453

Ahmad A, Khan M, Paul A, Din S, Rathore MM, Jeon G, Choi GS (2017) Toward modeling and optimization of features selection in big data based social internet of things. Future Gener Comput Syst

Ai W, Li K, Li K (2017) An effective hot topic detection method for microblog on spark. Appl Soft Comput

Amato F, Moscato V, Picariello A, Piccialli F (2017) SOS: a multimedia recommender system for online social networks. Future Gener Comput Syst

Apiletti D, Baralis E, Cerquitelli T, Garza P, Pulvirenti F, Michiardi P (2017) A parallel MapReduce algorithm to efficiently support itemset mining on high dimensional data. Big Data Res 10:53–69

Arias J, Gamez JA, Puerta JM (2017) Learning distributed discrete Bayesian network classifiers under MapReduce with Apache spark. Knowl-Based Syst 117:16–26

Aufaure MA, Chiky R, Curé O, Khrouf H, Kepeklian G (2016) From business intelligence to semantic data stream management. Future Gener Comput Syst 63:100–107

Babar M, Arif F (2017) Smart urban planning using big data analytics to contend with the interoperability in Internet of Things. Future Gener Comput Syst 77:65–76
Barba-González C, García-Nieto J, Nebro AJ, Cordero JA, Durillo JJ, Navas-Delgado I, Aldana-Montes JF (2017) jMetalSP: a framework for dynamic multi-objective big data optimization. Appl Soft Comput
Basanta-Val P, Fernández-García N, Wellings AJ, Audsley NC (2015) Improving the predictability of distributed stream processors. Future Gener Comput Syst 52:22–36
Basanta-Val, P., Fernández-García, N., & Sánchez-Fernández, L. (2017). Predictable remote invocations for distributed stream processing. Future Gener Comput Syst
Batarseh FA, Latif EA (2016) Assessing the quality of service using big data analytics: with application to healthcare. Big Data Res 4:13–24
Bechini A, Marcelloni F, Segatori A (2016) A MapReduce solution for associative classification of big data. Inf Sci 332:33–55
Bei Z, Yu Z, Luo N, Jiang C, Xu C, Feng S (2018) Configuring in-memory cluster computing using random forest. Future Gener Comput Syst 79:1–15
Bharti SK, Vachha B, Pradhan RK, Babu KS, Jena SK (2016) Sarcastic sentiment detection in tweets streamed in real time: a big data approach. Digital Commun Netw 2(3):108–121
Carcillo F, Dal Pozzolo A, Le Borgne YA, Caelen O, Mazzer Y, Bontempi G (2018) Scarff: a scalable framework for streaming credit card fraud detection with spark. Inf Fusion 41:182–194
Castiglione A, Colace F, Moscato V, Palmieri F (2017) CHIS: a big data infrastructure to manage digital cultural items. Future Gener Comput Syst
Chen CP, Zhang CY (2014) Data-intensive applications, challenges, techniques and technologies: a survey on big data. Inf Sci 275:314–347
Chen G, Wu S, Wang Y (2015) The evolvement of big data systems: from the perspective of an information security application, 65–73
Chen H, Li T, Cai Y, Luo C, Fujita H (2016) Parallel attribute reduction in dominance-based neighborhood rough set. Inf Sci 373:351–368
Chen Y, Crespi N, Ortiz AM, Shu L (2017) Reality mining: a prediction algorithm for disease dynamics based on mobile big data. Inf Sci 379:82–93
De Maio C, Fenza G, Loia V, Orciuoli F (2017) Distributed online temporal fuzzy concept analysis for stream processing in smart cities. J Parallel Distrib Comput 110:31–41
Del Río S, López V, Benítez JM, Herrera F (2014) On the use of MapReduce for imbalanced big data using random forest. Inf Sci 285:112–137
Ding L, Liu Y, Han B, Zhang S, Song B (2017) HB-file: an efficient and effective high-dimensional big data storage structure based on US-ELM. Neurocomputing 261:184–192
Eiras-Franco C, Bolón-Canedo V, Ramos S, González-Domínguez J, Alonso-Betanzos A, Touriño J (2016) Multithreaded and spark parallelization of feature selection filters. J Comput Sci 17:609–619
Elkano M, Galar M, Sanz J, Bustince H (2017) CHI-BD: a fuzzy rule-based classification system for big data classification problems. Fuzzy Sets Syst
Elsebakhi E, Lee F, Schendel E, Haque A, Kathireason N, Pathare T, Al-Ali R (2015) Large-scale machine learning based on functional networks for biomedical big data with high performance computing platforms. J Comput Sci 11:69–81
Fernández-Rodríguez JY, Álvarez-García JA, Fisteus JA, Luaces MR, Magaña VC (2017) Benchmarking real-time vehicle data streaming models for a Smart City. Inf Syst 72:62–76
Ferranti A, Marcelloni F, Segatori A, Antonelli M, Ducange P (2017) A distributed approach to multi-objective evolutionary generation of fuzzy rule-based classifiers from big data. Inf Sci 415:319–340
Fonseca A, Cabral B (2017) Prototyping a GPGPU neural network for deep-learning big data analysis. Big Data Res 8:50–56
Gadiraju KK, Verma M, Davis KC, Talaga PG (2016) Benchmarking performance for migrating a relational application to a parallel implementation. Future Gener Comput Syst 63:148–156
Genuer R, Poggi JM, Tuleau-Malot C, Villa-Vialaneix N (2017) Random forests for big data. Big Data Res 9:28–46
Ghadiri N, Ghaffari M, Nikbakht MA (2016) BigFCM: fast, precise and scalable FCM on Hadoop. arXiv preprint arXiv:1605.03047
Guo J, Song B, Yu FR, Yan Z, Yang LT (2017) Object detection among multimedia big data in the compressive measurement domain under mobile distributed architecture. Future Gener Comput Syst 76:519–527
He W, Wu H, Yan G, Akula V, Shen J (2015) A novel social media competitive analytics framework with sentiment benchmarks. Inf Manag 52(7):801–812
Hernández ÁB, Perez MS, Gupta S, Muntés-Mulero V (2017) Using machine learning to optimize parallelism in big data applications. Future Gener Comput Syst

Hidalgo N, Wladdimiro D, Rosas E (2017) Self-adaptive processing graph with operator fission for elastic stream processing. J Syst Softw 127:205–216
Higashino WA, Capretz MA, Bittencourt LF (2016) CEPSim: modelling and simulation of complex event processing systems in cloud environments. Future Gener Comput Syst 65:122–139
Huang S, Wang B, Qiu J, Yao J, Wang G, Yu G (2016) Parallel ensemble of online sequential extreme learning machine based on map reduce. Neurocomputing 174:352–367
Huang CS, Tsai MF, Huang PH, Su LD, Lee KS (2017) Distributed asteroid discovery system for large astronomical data. J Netw Comput Appl 93:27–37
Iqbal R, Doctor F, More B, Mahmud S, Yousuf U (2017) Big data analytics and computational intelligence for cyber–physical systems: recent trends and state of the art applications. Future Gener Comput Syst
Jayasena KPN, Li L, Xie Q (2017) Multi-modal multimedia big data analyzing architecture and resource allocation on cloud platform. Neurocomputing 253:135–143
Jiang R, Lu R, Choo KKR (2018) Achieving high performance and privacy-preserving query over encrypted multidimensional big metering data. Future Gener Comput Syst 78:392–401
Karunaratne P, Karunasekera S, Harwood A (2017) Distributed stream clustering using micro-clusters on Apache Storm. J Parallel Distrib Comput 108:74–84
Kousiouris G, Akbar A, Sancho J, Ta-shma P, Psychas A, Kyriazis D, Varvarigou T (2018) An integrated information lifecycle management framework for exploiting social network data to identify dynamic large crowd concentration events in smart cities applications. Future Gener Comput Syst 78:516–530
Kovalchuk SV, Krotov E, Smirnov PA, Nasonov DA, Yakovlev AN (2018) Distributed data-driven platform for urgent decision making in cardiological ambulance control. Future Gener Comput Syst 79:144–154
Kranjc J, Orač R, Podpečan V, Lavrač N, Robnik-Šikonja M (2017) ClowdFlows: online workflows for distributed big data mining. Future Gener Comput Syst 68:38–58
Kumar M, Rath SK (2015) Classification of microarray using MapReduce based proximal support vector machine classifier. Knowl-Based Syst 89:584–602
Liang Y, Wu D, Liu G, Li Y, Gao C, Ma ZJ, Wu W (2016) Big data-enabled multiscale serviceability analysis for aging bridges. Digit Commun Netw 2(3):97–107
Lin W, Dou W, Zhou Z, Liu C (2015) A cloud-based framework for home-diagnosis service over big medical data. J Syst Softw 102:192–206
Maillo J, Ramírez S, Triguero I, Herrera F (2017) kNN-IS: an iterative spark-based design of the k-nearest neighbors classifier for big data. Knowl-Based Syst 117:3–15
Manco G, Ritacco E, Rullo P, Gallucci L, Astill W, Kimber D, Antonelli M (2017) Fault detection and explanation through big data analysis on sensor streams. Expert Syst Appl 87:141–156
Manogaran G, Varatharajan R, Lopez D, Kumar PM, Sundarasekar R, Thota C (2017) A new architecture of internet of things and big data ecosystem for secured smart healthcare monitoring and alerting system. Future Gener Comput Syst
Maté A, Peral J, Ferrández A, Gil D, Trujillo J (2016) A hybrid integrated architecture for energy consumption prediction. Future Gener Comput Syst 63:131–147
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
Mestre DG, Pires CES, Nascimento DC (2017) Towards the efficient parallelization of multi-pass adaptive blocking for entity matching. J Parallel Distrib Comput 101:27–40
Mohapatra SK, Sahoo PK, Wu SL (2016a) Big data analytic architecture for intruder detection in heterogeneous wireless sensor networks. J Netw Comput Appl 66:236–249
Mohapatra SK, Sahoo PK, Wu SL (2016b) Big data analytic architecture for intruder detection in heterogeneous wireless sensor networks. J Netw Comput Appl 66:236–249
Nair LR, Shetty SD, Shetty SD (2017) Applying spark based machine learning model on streaming big data for health status prediction. Comput Electr Eng
Najafabadi MK, Mahrin MNR (2016) A systematic literature review on the state of research and practice of collaborative filtering technique and implicit feedback. Artif Intell Rev 45(2):167–201
Najafabadi MK, Mohamed AH, Mahrin MNR (2017) A survey on data mining techniques in recommender systems. Soft Comput, 1–28
Najafabadi MK, Mahrin MNR, Chuprat S, Sarkan HM (2017b) Improving the accuracy of collaborative filtering recommendations using clustering and association rules mining on implicit data. Comput Hum Behav 67:113–128
Nghiem PP, Figueira SM (2016) Towards efficient resource provisioning in MapReduce. J Parallel Distrib Comput 95:29–41
Nguyen T, Larsen ME, O'Dea B, Nguyen DT, Yearwood J, Phung D, Christensen H (2017) Kernel-based features for predicting population health indices from geocoded social media data. Decis Support Syst 102:22–31

Oneto L, Fumeo E, Clerico G, Canepa R, Papa F, Dambra C, Anguita D (2017) Train delay prediction systems: a big data analytics perspective. Big Data Res

Pedersen E, Bongo LA (2017) Large-scale biological meta-database management. Future Gener Comput Syst 67:481–489

Peralta D, García S, Benitez JM, Herrera F (2017) Minutiae-based fingerprint matching decomposition: methodology for big data frameworks. Inf Sci 408:198–212

Plimpton SJ, Shead T (2014) Streaming data analytics via message passing with application to graph algorithms. J Parallel Distrib Comput 74(8):2687–2698

Prajapati DJ, Garg S, Chauhan NC (2017) MapReduce based multilevel consistent and inconsistent association rule detection from big data using interestingness measures. Big Data Res 9:18–27

Pulgar-Rubio F, Rivera-Rivas AJ, Pérez-Godoy MD, González P, Carmona CJ, del Jesus MJ (2017) MEFASD-BD: multi-objective evolutionary fuzzy algorithm for subgroup discovery in big data environments-A MapReduce solution. Knowl-Based Syst 117:70–78

Qian J, Lv P, Yue X, Liu C, Jing Z (2015) Hierarchical attribute reduction algorithms for big data using MapReduce. Knowl-Based Syst 73:18–31

Rahman MN, Esmailpour A, Zhao J (2016) Machine learning with big data an efficient electricity generation forecasting system. Big Data Res 5:9–15

Rathore MM, Ahmad A, Paul A, Rho S (2016) Urban planning and building smart cities based on the internet of things using big data analytics. Comput Netw 101:63–80

Rathore MM, Paul A, Ahmad A, Chilamkurthi N, Hong WH, Seo H (2017) Real-time secure communication for Smart City in high-speed big data environment. Future Gener Comput Syst

Ruan G, Zhang H (2017) Closed-loop big data analysis with visualization and scalable computing. Big Data Res 8:12–26

Sahal R, Khafagy MH, Omara FA (2017) Exploiting coarse-grained reused-based opportunities in Big Data multi-query optimization. J Comput Sci

Singh H, Bawa S (2017) A MapReduce-based scalable discovery and indexing of structured big data. Future Gener Comput Syst 73:32–43

Singh K, Guntuku SC, Thakur A, Hota C (2014) Big data analytics framework for peer-to-peer botnet detection using random forests. Inf Sci 278:488–497

Singh S, Garg R, Mishra PK (2017) Performance optimization of MapReduce-based Apriori algorithm on Hadoop cluster. Comput Electr Eng

Spivak A, Razumovskiy A, Nasonov D, Boukhanovsky A, Redice A (2018) Storage tier-aware replicative data reorganization with prioritization for efficient workload processing. Future Gener Comput Syst 79:618–629

Sun D, Zhang G, Yang S, Zheng W, Khan SU, Li K (2015) Re-stream: real-time and energy-efficient resource scheduling in big data stream computing environments. Inf Sci 319:92–112

Tennant M, Stahl F, Rana O, Gomes JB (2017) Scalable real-time classification of data streams with concept drift. Future Gener Comput Syst 75:187–199

Triguero I, Peralta D, Bacardit J, García S, Herrera F (2015) MRPR: a MapReduce solution for prototype reduction in big data classification. Neurocomputing 150:331–345

Tripathy BK, Mittal D (2016) Hadoop based uncertain possibilistic kernelized c-means algorithms for image segmentation and a comparative analysis. Appl Soft Comput 46:886–923

Tsai CW, Liu SJ, Wang YC (2017) A parallel metaheuristic data clustering framework for cloud. J Parallel Distrib Comput

Um JH, Lee S, Kim TH, Jeong CH, Song SK, Jung H (2016) Semantic complex event processing model for reasoning research activities. Neurocomputing 209:39–45

Vennila V, Kannan AR (2016) Symmetric matrix-based predictive classifier for big data computation and information sharing in cloud. Comput Electr Eng 56:831–841

Wang H, Belhassena A (2017) Parallel trajectory search based on distributed index. Inf Sci 388:62–83

Wang B, Huang S, Qiu J, Liu Y, Wang G (2015) Parallel online sequential extreme learning machine based on MapReduce. Neurocomputing 149:224–232

Wang H, Xu Z, Fujita H, Liu S (2016) Towards felicitous decision making: an overview on challenges and trends of big data. Inf Sci 367:747–765

Wang J, He C, Liu Y, Tian G, Peng I, Xing J, Wang FL (2017a) Efficient alarm behavior analytics for telecom networks. Inf Sci 402:1–14

Wang, Y., Geng, S., & Gao, H. (2017). A proactive decision support method based on deep reinforcement learning and state partition. Knowl-Based Syst

Xia Y, Chen J, Lu X, Wang C, Xu C (2016) Big traffic data processing framework for intelligent monitoring and recording systems. Neurocomputing 181:139–146

Yuan J, Chen M, Jiang T, Li T (2017) Complete tolerance relation based parallel filling for incomplete energy big data. Knowl-Based Syst 132:215–225

Zhang F, Cao J, Khan SU, Li K, Hwang K (2015) A task-level adaptive MapReduce framework for real-time streaming data in healthcare applications. Future Gener Comput Syst 43:149–160

Zhang CY, Chen CP, Chen D, Ng KT (2016) MapReduce based distributed learning algorithm for restricted Boltzmann machine. Neurocomputing 198:4–11

Affiliations

Azlinah Mohamed[1] · Maryam Khanian Najafabadi[2] · Yap Bee Wah[1] · Ezzatul Akmal Kamaru Zaman[1] · Ruhaila Maskat[1]

[1] Advanced Analytics Engineering Centre, Faculty Computer and Mathematical Sciences, Universiti Teknologi MARA (UiTM), Shah Alam, Selangor, Malaysia

[2] Present Address: Faculty of Information Technology, INTI International University & Colleges, Nilai, Malaysia