Tsai *et al. Journal of Big Data (2015) 2:21*
DOI 10.1186/s40537-015-0030-3

Journal of Big Data
a SpringerOpen Journal

SURVEY PAPER Open Access

Big data analytics: a survey

Chun-Wei Tsai[1], Chin-Feng Lai[2], Han-Chieh Chao[1,3,4] and Athanasios V. Vasilakos[5*]

*Correspondence:
th.vasilakos@gmail.com
[5] Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, SE-931 87 Skellefteå, Sweden
Full list of author information is available at the end of the article

Abstract

The age of big data is now coming. But the traditional data analytics may not be able to handle such large quantities of data. The question that arises now is, how to develop a high performance *platform* to efficiently analyze big data and how to design an appropriate *mining algorithm* to find the useful things from big data. To deeply discuss this issue, this paper begins with a brief introduction to data analytics, followed by the discussions of big data analytics. Some important open issues and further research directions will also be presented for the next step of big data analytics.

Keywords: Big data, data analytics, data mining

Introduction

As the information technology spreads fast, most of the data were born digital as well as exchanged on internet today. According to the estimation of Lyman and Varian [1], the new data stored in digital media devices have already been more than 92 % in 2002, while the size of these new data was also more than five exabytes. In fact, the problems of analyzing the large scale data were not suddenly occurred but have been there for several years because the creation of data is usually much easier than finding useful things from the data. Even though computer systems today are much faster than those in the 1930s, the large scale data is a strain to analyze by the computers we have today.

In response to the problems of analyzing *large-scale data*, quite a few efficient methods [2], such as sampling, data condensation, density-based approaches, grid-based approaches, divide and conquer, incremental learning, and distributed computing, have been presented. Of course, these methods are constantly used to improve the performance of the operators of data analytics process.[1] The results of these methods illustrate that with the efficient methods at hand, we may be able to analyze the large-scale data in a reasonable time. The dimensional reduction method (e.g., principal components analysis; PCA [3]) is a typical example that is aimed at reducing the input data volume to accelerate the process of data analytics. Another reduction method that reduces the data computations of data clustering is sampling [4], which can also be used to speed up the computation time of data analytics.

Although the advances of computer systems and internet technologies have witnessed the development of computing hardware following the Moore's law for several decades,

[1] In this paper, by the data analytics, we mean the whole KDD process, while by the data analysis, we mean the part of data analytics that is aimed at finding the hidden information in the data, such as data mining.

the problems of handling the large-scale data still exist when we are entering the age of *big data*. That is why Fisher et al. [5] pointed out that big data means that the data is unable to be handled and processed by most current information systems or methods because data in the big data era will not only become too big to be loaded into a single machine, it also implies that most traditional data mining methods or data analytics developed for a centralized data analysis process may not be able to be applied directly to big data. In addition to the issues of data size, Laney [6] presented a well-known definition (also called 3Vs) to explain what is the "big" data: volume, velocity, and variety. The definition of 3Vs implies that the data size is large, the data will be created rapidly, and the data will be existed in multiple types and captured from different sources, respectively. Later studies [7, 8] pointed out that the definition of 3Vs is insufficient to explain the big data we face now. Thus, veracity, validity, value, variability, venue, vocabulary, and vagueness were added to make some complement explanation of big data [8].

The report of IDC [9] indicates that the marketing of big data is about $16.1 billion in 2014. Another report of IDC [10] forecasts that it will grow up to $32.4 billion by 2017. The reports of [11] and [12] further pointed out that the marketing of big data will be $46.34 billion and $114 billion by 2018, respectively. As shown in Fig. 1, even though the marketing values of big data in these researches and technology reports [9–15] are different, these forecasts usually indicate that the scope of big data will be grown rapidly in the forthcoming future.

In addition to marketing, from the results of disease control and prevention [16], business intelligence [17], and smart city [18], we can easily understand that big data is of vital importance everywhere. A numerous researches are therefore focusing on developing effective technologies to analyze the big data. To discuss in deep the big data analytics, this paper gives not only a systematic description of traditional large-scale data analytics but also a detailed discussion about the differences between data and big data analytics framework for the data scientists or researchers to focus on the big data analytics.

Moreover, although several data analytics and frameworks have been presented in recent years, with their pros and cons being discussed in different studies, a complete discussion from the perspective of data mining and knowledge discovery in databases still is needed. As a result, this paper is aimed at providing a brief review for the

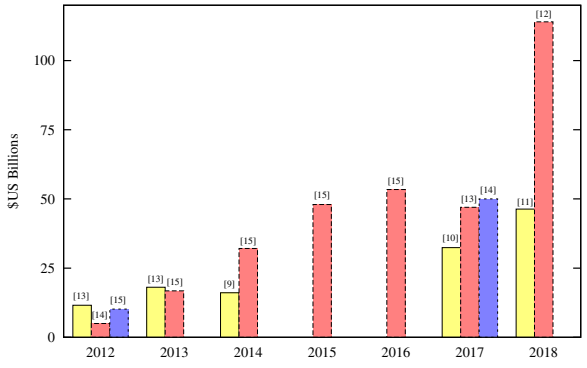


Fig. 1 Expected trend of the marketing of big data between 2012 and 2018. Note that *yellow*, *red*, and *blue* of different colored box represent the order of appearance of reference in this paper for particular year

researchers on the data mining and distributed computing domains to have a basic idea to use or develop data analytics for big data.

Figure 2 shows the roadmap of this paper, and the remainder of the paper is organized as follows. "Data analytics" begins with a brief introduction to the data analytics, and then "Big data analytics" will turn to the discussion of big data analytics as well as state-of-the-art data analytics algorithms and frameworks. The open issues are discussed in "The open issues" while the conclusions and future trends are drawn in "Conclusions".

Data analytics

To make the whole process of knowledge discovery in databases (KDD) more clear, Fayyad and his colleagues summarized the KDD process by a few operations in [19], which are selection, preprocessing, transformation, data mining, and interpretation/evaluation. As shown in Fig. 3, with these operators at hand we will be able to build a complete data analytics system to gather data first and then find information from the data and display the knowledge to the user. According to our observation, the number of research articles and technical reports that focus on data mining is typically more than the number focusing on other operators, but it does not mean that the other operators of KDD are unimportant. The other operators also play the vital roles in KDD process because they will strongly impact the final result of KDD. To make the discussions on the main operators of KDD process more concise, the following sections will focus on those depicted in Fig. 3, which were simplified to three parts (input, data analytics, and output) and seven operators (gathering, selection, preprocessing, transformation, data mining, evaluation, and interpretation).

Data input

As shown in Fig. 3, the gathering, selection, preprocessing, and transformation operators are in the input part. The selection operator usually plays the role of knowing which

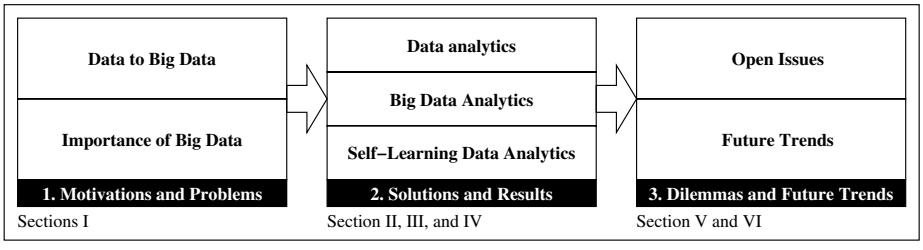


Fig. 2 Roadmap of this paper

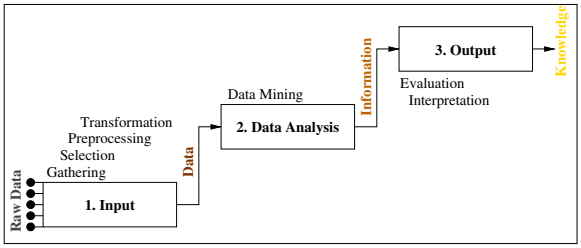


Fig. 3 The process of knowledge discovery in databases

kind of data was required for data analysis and select the relevant information from the gathered data or databases; thus, these gathered data from different data resources will need to be integrated to the target data. The preprocessing operator plays a different role in dealing with the input data which is aimed at detecting, cleaning, and filtering the unnecessary, inconsistent, and incomplete data to make them the useful data. After the selection and preprocessing operators, the characteristics of the secondary data still may be in a number of different data formats; therefore, the KDD process needs to transform them into a data-mining-capable format which is performed by the transformation operator. The methods for reducing the complexity and downsizing the data scale to make the data useful for data analysis part are usually employed in the transformation, such as dimensional reduction, sampling, coding, or transformation.

The data extraction, data cleaning, data integration, data transformation, and data reduction operators can be regarded as the preprocessing processes of data analysis [20] which attempts to extract useful data from the raw data (also called the primary data) and refine them so that they can be used by the following data analyses. If the data are a duplicate copy, incomplete, inconsistent, noisy, or outliers, then these operators have to clean them up. If the data are too complex or too large to be handled, these operators will also try to reduce them. If the raw data have errors or omissions, the roles of these operators are to identify them and make them consistent. It can be expected that these operators may affect the analytics result of KDD, be it positive or negative. In summary, the systematic solutions are usually to reduce the complexity of data to accelerate the computation time of KDD and to improve the accuracy of the analytics result.

Data analysis

Since the data analysis (as shown in Fig. 3) in KDD is responsible for finding the hidden patterns/rules/information from the data, most researchers in this field use the term data mining to describe how they refine the "ground" (i.e, raw data) into "gold nugget" (i.e., information or knowledge). The data mining methods [20] are not limited to data problem specific methods. In fact, other technologies (e.g., statistical or machine learning technologies) have also been used to analyze the data for many years. In the early stages of data analysis, the statistical methods were used for analyzing the data to help us understand the situation we are facing, such as public opinion poll or TV programme rating. Like the statistical analysis, the problem specific methods for data mining also attempted to understand the meaning from the collected data.

After the data mining problem was presented, some of the domain specific algorithms are also developed. An example is the apriori algorithm [21] which is one of the useful algorithms designed for the association rules problem. Although most definitions of data mining problems are simple, the computation costs are quite high. To speed up the response time of a data mining operator, machine learning [22], metaheuristic algorithms [23], and distributed computing [24] were used alone or combined with the traditional data mining algorithms to provide more efficient ways for solving the data mining problem. One of the well-known combinations can be found in [25], Krishna and Murty attempted to combine genetic algorithm and *k*-means to get better clustering result than *k*-means alone does.

As Fig. 4 shows, most data mining algorithms contain the initialization, data input and output, data scan, rules construction, and rules update operators [26]. In Fig. 4, *D*

```
Input data D
Initialize candidate solutions r
While the termination criterion is not met
    d = Scan(D)
    v = Construct(d, r, o)
    r = Update(v)
End
Output rules r
```

Fig. 4 Data mining algorithm

represents the raw data, d the data from the scan operator, r the rules, o the predefined measurement, and v the candidate rules. The scan, construct, and update operators will be performed repeatedly until the termination criterion is met. The timing to employ the scan operator depends on the design of the data mining algorithm; thus, it can be considered as an optional operator. Most of the data algorithms can be described by Fig. 4 in which it also shows that the representative algorithms—*clustering*, *classification*, *association rules*, and *sequential patterns*—will apply these operators to find the hidden information from the raw data. Thus, modifying these operators will be one of the possible ways for enhancing the performance of the data analysis.

Clustering is one of the well-known data mining problems because it can be used to understand the "new" input data. The basic idea of this problem [27] is to separate a set of unlabeled input data[2] to k different groups, e.g., such as k-means [28]. Classification [20] is the opposite of clustering because it relies on a set of labeled input data to construct a set of classifiers (i.e., groups) which will then be used to classify the unlabeled input data to the groups to which they belong. To solve the classification problem, the decision tree-based algorithm [29], naïve Bayesian classification [30], and support vector machine (SVM) [31] are widely used in recent years.

Unlike clustering and classification that attempt to classify the input data to k groups, association rules and sequential patterns are focused on finding out the "relationships" between the input data. The basic idea of association rules [21] is find all the co-occurrence relationships between the input data. For the association rules problem, the apriori algorithm [21] is one of the most popular methods. Nevertheless, because it is computationally very expensive, later studies [32] have attempted to use different approaches to reducing the cost of the apriori algorithm, such as applying the genetic algorithm to this problem [33]. In addition to considering the relationships between the input data, if we also consider the sequence or time series of the input data, then it will be referred to as the sequential pattern mining problem [34]. Several apriori-like algorithms were presented for solving it, such as generalized sequential pattern [34] and sequential pattern discovery using equivalence classes [35].

Output the result

Evaluation and interpretation are two vital operators of the output. Evaluation typically plays the role of measuring the results. It can also be one of the operators for the data

[2] In this paper, by an unlabeled input data, we mean that it is unknown to which group the input data belongs. If all the input data are unlabeled, it means that the distribution of the input data is unknown.

mining algorithm, such as the sum of squared errors which was used by the selection operator of the genetic algorithm for the clustering problem [25].

To solve the data mining problems that attempt to classify the input data, two of the major goals are: (1) cohesion—the distance between each data and the centroid (mean) of its cluster should be as small as possible, and (2) coupling—the distance between data which belong to different clusters should be as large as possible. In most studies of data clustering or classification problems, the sum of squared errors (SSE), which was used to measure the cohesion of the data mining results, can be defined as

$$\mathrm{SSE} = \sum_{i=1}^{k} \sum_{j=1}^{n_i} D(x_{ij} - c_i), \tag{1}$$

$$c_i = \frac{1}{n_i} \sum_{j=1}^{n_i} x_{ij}, \tag{2}$$

where k is the number of clusters which is typically given by the user; n_i the number of data in the ith cluster; x_{ij} the jth datum in the ith cluster; c_i is the mean of the ith cluster; and $n = \sum_{i=1}^{k} n_i$ is the number of data. The most commonly used distance measure for the data mining problem is the Euclidean distance, which is defined as

$$D(p_i, p_j) = \left(\sum_{l=1}^{d} |p_{il}, p_{jl}|^2 \right)^{1/2}, \tag{3}$$

where p_i and p_j are the positions of two different data. For solving different data mining problems, the distance measurement $D(p_i, p_j)$ can be the Manhattan distance, the Minkowski distance, or even the cosine similarity [36] between two different documents.

Accuracy (ACC) is another well-known measurement [37] which is defined as

$$\mathrm{ACC} = \frac{\text{Number of cases correctly classified}}{\text{Total number of test cases}}. \tag{4}$$

To evaluate the classification results, precision (p), recall (r), and F-measure can be used to measure how many data that do not belong to group A are incorrectly classified into group A; and how many data that belong to group A are not classified into group A. A simple confusion matrix of a classifier [37] as given in Table 1 can be used to cover all the situations of the classification results. In Table 1, TP and TN indicate the numbers of positive examples and negative examples that are correctly classified, respectively; FN and FP indicate the numbers of positive examples and negative examples that are incorrectly classified, respectively. With the confusion matrix at hand, it is much easier to describe the meaning of precision (p), which is defined as

$$p = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \tag{5}$$

and the meaning of recall (r), which is defined as

$$r = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}. \tag{6}$$

Table 1 Confusion matrix of a classifier [37]

	Classified positive	Classified negative
Actual positive	TP	FN
Actual negative	FP	TN

The *F*-measure can then be computed as

$$F = \frac{2pr}{p+r}. \tag{7}$$

In addition to the above-mentioned measurements for evaluating the data mining results, the computation cost and response time are another two well-known measurements. When two different mining algorithms can find the same or similar results, of course, how fast they can get the final mining results will become the most important research topic.

After *something* (e.g., classification rules) is found by data mining methods, the two essential research topics are: (1) the work to navigate and explore the meaning of the results from the data analysis to further support the user to do the applicable decision can be regarded as the interpretation operator [38], which in most cases, gives useful interface to display the information [39] and (2) a meaningful summarization of the mining results [40] can be made to make it easier for the user to understand the information from the data analysis. The data summarization is generally expected to be one of the simple ways to provide a concise piece of information to the user because human has trouble of understanding vast amounts of complicated information. A simple data summarization can be found in the clustering search engine, when a query "oasis" is sent to Carrot2 (http://search.carrot2.org/stable/search), it will return some keywords to represent each group of the clustering results for web links to help us recognize which category needed by the user, as shown in the left side of Fig. 5.

A useful graphical user interface is another way to provide the meaningful information to an user. As explained by Shneiderman in [39], we need "overview first, zoom and filter, then retrieve the details on demand". The useful graphical user interface [38, 41] also makes it easier for the user to comprehend the meaning of the results when the number of dimensions is higher than three. How to display the results of data mining will affect the user's perspective to make the decision. For instance, data mining can help us find "type A influenza" at a particular region, but without the time series and flu virus infected information of patients, the government could not recognize what situation (pandemic or controlled) we are facing now so as to make appropriate responses to that. For this reason, a better solution to merge the information from different sources and mining algorithm results will be useful to let the user make the right decision.

Summary

Since the problems of handling and analyzing large-scale and complex input data always exist in data analytics, several efficient analysis methods were presented to accelerate the computation time or to reduce the memory cost for the KDD process, as shown in Table 2. The study of [42] shows that the basic mathematical concepts (i.e., triangle

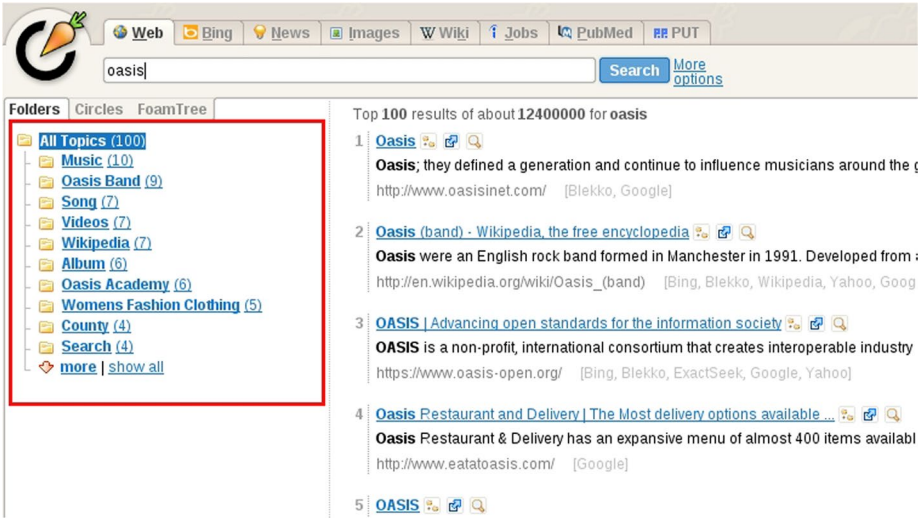


Fig. 5 Screenshot of the results of clustering search engine

inequality) can be used to reduce the computation cost of a clustering algorithm. Another study [43] shows that the new technologies (i.e., distributed computing by GPU) can also be used to reduce the computation time of data analysis method. In addition to the well-known improved methods for these analysis methods (e.g., triangle inequality or distributed computing), a large proportion of studies designed their efficient methods based on the characteristics of mining algorithms or problem itself, which can be found in [32, 44, 45], and so forth. This kind of improved methods typically was designed for solving the drawback of the mining algorithms or using different ways to solve the mining problem. These situations can be found in most association rules and sequential patterns problems because the original assumption of these problems is for the analysis of large-scale dataset. Since the earlier frequent pattern algorithm (e.g., apriori algorithm) needs to scan the whole dataset many times which is computationally very expensive. How to reduce the number of times the whole dataset is scanned so as to save the computation cost is one of the most important things in all the frequent pattern studies. The similar situation also exists in data clustering and classification studies because the design concept of earlier algorithms, such as mining the patterns on-the-fly [46], mining partial patterns at different stages [47], and reducing the number of times the whole dataset is scanned [32], are therefore presented to enhance the performance of these mining algorithms. Since some of the data mining problems are NP-hard [48] or the solution space is very large, several recent studies [23, 49] have attempted to use metaheuristic algorithm as the mining algorithm to get the approximate solution within a reasonable time.

Abundant research results of data analysis [20, 27, 63] show possible solutions for dealing with the dilemmas of data mining algorithms. It means that the open issues of data analysis from the literature [2, 64] usually can help us easily find the possible solutions. For instance, the clustering result is extremely sensitive to the initial means, which can be mitigated by using multiple sets of initial means [65]. According to our observation, most data analysis methods have limitations for big data, that can be described as follows:

Table 2 Efficient data analytics methods for data mining

Problem	Method	References
Clustering	BIRCH	[44]
	DBSCAN	[45]
	Incremental DBSCAN	[46]
	RKM	[47]
	TKM	[42]
Classification	SLIQ	[50]
	TLAESA	[51]
	FastNN	[52]
	SFFS	[53]
	GPU-based SVM	[43]
Association rules	CLOSET	[54]
	FP-tree	[32]
	CHARM	[55]
	MAFIA	[56]
	FAST	[57]
Sequential patterns	SPADE	[58]
	CloSpan	[59]
	PrefixSpan	[60]
	SPAM	[61]
	ISE	[62]

- *Unscalability and centralization* Most data analysis methods are not for large-scale and complex dataset. The traditional data analysis methods cannot be scaled up because their design does not take into account large or complex datasets. The design of traditional data analysis methods typically assumed they will be performed in a single machine, with all the data in memory for the data analysis process. For this reason, the performance of traditional data analytics will be limited in solving the volume problem of big data.
- *Non-dynamic* Most traditional data analysis methods cannot be dynamically adjusted for different situations, meaning that they do not analyze the input data on-the-fly. For example, the classifiers are usually fixed which cannot be automatically changed. The incremental learning [66] is a promising research trend because it can dynamically adjust the the classifiers on the training process with limited resources. As a result, the performance of traditional data analytics may not be useful to the problem of velocity problem of big data.
- *Uniform data structure* Most of the data mining problems assume that the format of the input data will be the same. Therefore, the traditional data mining algorithms may not be able to deal with the problem that the formats of different input data may be different and some of the data may be incomplete. How to make the input data from different sources the same format will be a possible solution to the variety problem of big data.

Because the traditional data analysis methods are not designed for large-scale and complex data, they are almost impossible to be capable of analyzing the big data. Redesigning and changing the way the data analysis methods are designed are two critical

trends for big data analysis. Several important concepts in the design of the big data analysis method will be given in the following sections.

Big data analytics

Nowadays, the data that need to be analyzed are not just large, but they are composed of various data types, and even including streaming data [67]. Since big data has the unique features of "massive, high dimensional, heterogeneous, complex, unstructured, incomplete, noisy, and erroneous," which may change the statistical and data analysis approaches [68]. Although it seems that big data makes it possible for us to collect more data to find more useful information, the truth is that more data do not necessarily mean more useful information. It may contain more ambiguous or abnormal data. For instance, a user may have multiple accounts, or an account may be used by multiple users, which may degrade the accuracy of the mining results [69]. Therefore, several new issues for data analytics come up, such as privacy, security, storage, fault tolerance, and quality of data [70].

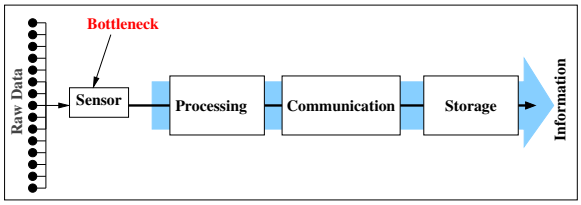
The big data may be created by handheld device, social network, internet of things, multimedia, and many other new applications that all have the characteristics of volume, velocity, and variety. As a result, the whole data analytics has to be re-examined from the following perspectives:

- From the volume perspective, the deluge of input data is the very first thing that we need to face because it may paralyze the data analytics. Different from traditional data analytics, for the wireless sensor network data analysis, Baraniuk [71] pointed out that the bottleneck of big data analytics will be shifted from sensor to processing, communications, storage of sensing data, as shown in Fig. 6. This is because sensors can gather much more data, but when uploading such large data to upper layer system, it may create bottlenecks everywhere.
- In addition, from the velocity perspective, real-time or streaming data bring up the problem of large quantity of data coming into the data analytics within a short duration but the device and system may not be able to handle these input data. This situation is similar to that of the network flow analysis for which we typically cannot mirror and analyze everything we can gather.
- From the variety perspective, because the incoming data may use different types or have incomplete data, how to handle them also bring up another issue for the input operators of data analytics.

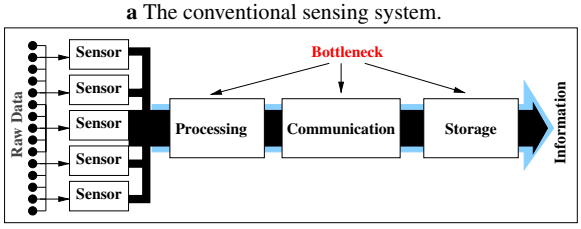
In this section, we will turn the discussion to the big data analytics process.

Big data input

The problem of handling a vast quantity of data that the system is unable to process is not a brand-new research issue; in fact, it appeared in several early approaches [2, 21, 72], e.g., marketing analysis, network flow monitor, gene expression analysis, weather forecast, and even astronomy analysis. This problem still exists in big data analytics today; thus, preprocessing is an important task to make the computer, platform, and analysis algorithm be able to handle the input data. The traditional data preprocessing



a The conventional sensing system.



b The sensing system for data deluge.

Fig. 6 The comparison between traditional data analysis and big data analysis on wireless sensor network

methods [73] (e.g., compression, sampling, feature selection, and so on) are expected to be able to operate effectively in the big data age. However, a portion of the studies still focus on how to reduce the complexity of the input data because even the most advanced computer technology cannot efficiently process the whole input data by using a single machine in most cases. By using domain knowledge to design the preprocessing operator is a possible solution for the big data. In [74], Ham and Lee used the domain knowledge, *B*-tree, divide-and-conquer to filter the unrelated log information for the mobile web log analysis. A later study [75] considered that the computation cost of preprocessing will be quite high for massive logs, sensor, or marketing data analysis. Thus, Dawelbeit and McCrindle employed the bin packing partitioning method to divide the input data between the computing processors to handle this high computations of preprocessing on cloud system. The cloud system is employed to preprocess the raw data and then output the refined data (e.g., data with uniform format) to make it easier for the data analysis method or system to preform the further analysis work.

Sampling and compression are two representative data reduction methods for big data analytics because reducing the size of data makes the data analytics computationally less expensive, thus faster, especially for the data coming to the system rapidly. In addition to making the sampling data represent the original data effectively [76], how many instances need to be selected for data mining method is another research issue [77] because it will affect the performance of the sampling method in most cases.

To avoid the application-level slow-down caused by the compression process, in [78], Jun et al. attempted to use the FPGA to accelerate the compression process. The I/O performance optimization is another issue for the compression method. For this reason, Zou et al. [79] employed the tentative selection and predictive dynamic selection and switched the appropriate compression method from two different strategies to improve the performance of the compression process. To make it possible for the compression method to efficiently compress the data, a promising solution is to apply the clustering method to the input data to divide them into several different groups and then compress these input data according to the clustering information. The compression method

described in [80] is one of this kind of solutions, it first clusters the input data and then compresses these input data via the clustering results while the study [81] also used clustering method to improve the performance of the compression process.

In summary, in addition to handling the large and fast data input, the research issues of heterogeneous data sources, incomplete data, and noisy data may also affect the performance of the data analysis. The input operators will have a stronger impact on the data analytics at the big data age than it has in the past. As a result, the design of big data analytics needs to consider how to make these tasks (e.g., data clean, data sampling, data compression) work well.

Big data analysis frameworks and platforms

Various solutions have been presented for the big data analytics which can be divided [82] into (1) Processing/Compute: Hadoop [83], Nvidia CUDA [84], or Twitter Storm [85], (2) Storage: Titan or HDFS, and (3) Analytics: MLPACK [86] or Mahout [87]. Although there exist commercial products for data analysis [83–86], most of the studies on the traditional data analysis are focused on the design and development of efficient and/or effective "ways" to find the useful things from the data. But when we enter the age of big data, most of the current computer systems will not be able to handle the whole dataset all at once; thus, how to design a good data analytics framework or platform[3] and how to design analysis methods are both important things for the data analysis process. In this section, we will start with a brief introduction to data analysis frameworks and platforms, followed by a comparison of them.

Researches in frameworks and platforms

To date, we can easily find tools and platforms presented by well-known organizations. The cloud computing technologies are widely used on these platforms and frameworks to satisfy the large demands of computing power and storage. As shown in Fig. 7, most of the works on KDD for big data can be moved to cloud system to speed up the response time or to increase the memory space. With the advance of these works, handling and analyzing big data within a reasonable time has become not so far away. Since the foundation functions to *handle* and *manage* the big data were developed gradually; thus, the data scientists nowadays do not have to take care of everything, from the raw data gathering to data analysis, by themselves if they use the existing platforms or technologies to handle and manage the data. The data scientists nowadays can pay more attention to *finding* out the useful information from the data even thought this task is typically like looking for a needle in a haystack. That is why several recent studies tried to present efficient and effective framework to analyze the big data, especially on find out the useful things.

Performance-oriented From the perspective of platform performance, Huai [88] pointed out that most of the traditional parallel processing models improve the performance of the system by using a new larger computer system to replace the old computer system, which is usually referred to as "scale up", as shown in Fig. 8a. But for the big data analytics, most researches improve the performance of the system by adding more

[3] In this paper, the analysis framework refers to the whole system, from raw data gathering, data reformat, data analysis, all the way to knowledge representation.

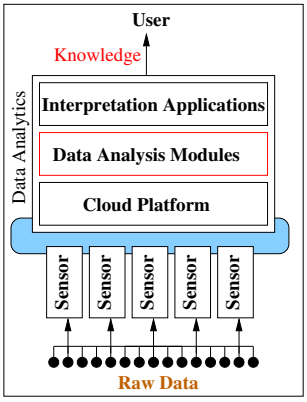


Fig. 7 The basic idea of big data analytics on cloud system

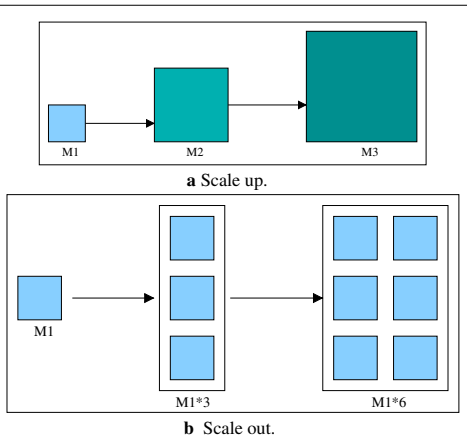


Fig. 8 The comparisons between scale up and scale out

similar computer systems to make it possible for a system to handle all the tasks that cannot be loaded or computed in a single computer system (called "scale out"), as shown in Fig. 8b where M1, M2, and M3 represent computer systems that have different computing power, respectively. For the scale up based solution, the computing power of the three systems is in the order of M3 > M2 > M1; but for the scale out based system, all we have to do is to keep adding more similar computer systems to to a system to increase its ability. To build a scalable and fault-tolerant manager for big data analysis, Huai et al. [88] presented a matrix model which consists of three matrices for data set (D), concurrent data processing operations (O), and data transformations (T), called DOT. The big data is divided into n subsets each of which is processed by a computer node (worker) in such a way that all the subsets are processed concurrently, and then the results from these n computer nodes are collected and transformed to a computer node. By using this framework, the whole data analysis framework is composed of several DOT blocks. The system performance can be easily enhanced by adding more DOT blocks to the system.

Another efficient big data analytics was presented in [89], called generalized linear aggregates distributed engine (GLADE). The GLADE is a multi-level tree-based data analytics system which consists of two types of computer nodes that are a coordinator

and workers. The simulation results [90] show that the GLADE can provide a better performance than Hadoop in terms of the execution time. Because Hadoop requires large memory and storage for data replication and it is a single master,[4] Essa et al. [91] presented a mobile agent based framework to solve these two problems, called the map reduce agent mobility (MRAM). The main reason is that each mobile agent can send its code and data to any other machine; therefore, the whole system will not be down if the master failed. Compared to Hadoop, the architecture of MRAM was changed from client/server to a distributed agent. The load time for MRAM is less than Hadoop even though both of them use the map-reduce solution and Java language. In [92], Herodotou et al. considered issues of the user needs and system workloads. They presented a self-tuning analytics system built on Hadoop for big data analysis. Since one of the major goals of their system is to adjust the system based on the user needs and system workloads to provide good performance automatically, the user usually does not need to understand and manipulate the Hadoop system. The study [93] was from the perspectives of data centric architecture and operational models to presented a big data architecture framework (BDAF) which includes: big data infrastructure, big data analytics, data structures and models, big data lifecycle management, and big data security. According to the observations of Demchenko et al. [93], cluster services, Hadoop related services, data analytics tools, databases, servers, and massively parallel processing databases are typically the required applications and services in big data analytics infrastructure.

Result-oriented Fisher et al. [5] presented a big data pipeline to show the workflow of big data analytics to extract the valuable knowledge from big data, which consists of the acquired data, choosing architecture, shaping data into architecture, coding/debugging, and reflecting works. From the perspectives of statistical computation and data mining, Ye et al. [94] presented an architecture of the services platform which integrates R to provide better data analysis services, called cloud-based big data mining and analyzing services platform (CBDMASP). The design of this platform is composed of four layers: the infrastructure services layer, the virtualization layer, the dataset processing layer, and the services layer. Several large-scale clustering problems (the datasets are of size from 0.1 G up to 25.6 G) were also used to evaluate the performance of the CBDMASP. The simulation results show that using map-reduce is much faster than using a single machine when the input data become too large. Although the size of the test dataset cannot be regarded as a big dataset, the performance of the big data analytics using map-reduce can be sped up via this kind of testings. In this study, map-reduce is a better solution when the dataset is of size more than 0.2 G, and a single machine is unable to handle a dataset that is of size more than 1.6 G.

Another study [95] presented a theorem to explain the big data characteristics, called HACE: the characteristics of big data usually are large-volume, Heterogeneous, Autonomous sources with distributed and decentralized control, and we usually try to find out some useful and interesting things from complex and evolving relationships of data. Based on these concerns and data mining issues, Wu and his colleagues [95] also presented a big data processing framework which includes data accessing and computing

[4] The whole system may be down when the master machine crashed for a system that has only one master.

tier, data privacy and domain knowledge tier, and big data mining algorithm tier. This work explains that the data mining algorithm will become much more important and much more difficult; thus, challenges will also occur on the design and implementation of big data analytics platform. In addition to the platform performance and data mining issues, the privacy issue for big data analytics was a promising research in recent years. In [96], Laurila et al. explained that the privacy is an essential problem when we try to find something from the data that are gathered from mobile devices; thus, data security and data anonymization should also be considered in analyzing this kind of data. Demirkan and Delen [97] presented a service-oriented decision support system (SODSS) for big data analytics which includes information source, data management, information management, and operations management.

Comparison between the frameworks/platforms of big data

In [98], Talia pointed out that cloud-based data analytics services can be divided into data analytics software as a service, data analytics platform as a service, and data analytics infrastructure as a service. A later study [99] presented a general architecture of big data analytics which contains multi-source big data collecting, distributed big data storing, and intra/inter big data processing. Since many kinds of data analytics frameworks and platforms have been presented, some of the studies attempted to compare them to give a guidance to choose the applicable frameworks or platforms for relevant works. To give a brief introduction to big data analytics, especially the platforms and frameworks, in [100], Cuzzocrea et al. first discuss how recent studies responded the "computational emergency" issue of big data analytics. Some open issues, such as data source heterogeneity and uncorrelated data filtering, and possible research directions are also given in the same study. In [101], Zhang and Huang used the 5Ws model to explain what kind of framework and method we need for different big data approaches. Zhang and Huang further explained that the 5Ws model represents *what* kind of data, *why* we have these data, *where* the data come from, *when* the data occur, *who* receive the data, and *how* the data are transferred. A later study [102] used the features (i.e., owner, workload, source code, low latency, and complexity) to compare the frameworks of Hadoop [83], Storm [85] and Drill [103]. Thus, it can be easily seen that the framework of Apache Hadoop has high latency compared with the other two frameworks. To better understand the strong and weak points of solutions of big data, Chalmers et al. [82] then employed the volume, variety, variability, velocity, user skill/experience, and infrastructure to evaluate eight solutions of big data analytics.

In [104], in addition to defining that a big data system should include data generation, data acquisition, data storage, and data analytics modules, Hu et al. also mentioned that a big data system can be decomposed into infrastructure, computing, and application layers. Moreover, a promising research for NoSQL storage systems was also discussed in this study which can be divided into *key-value*, *column*, *document*, and *row* databases. Since big data analysis is generally regarded as a high computation cost work, the high performance computing cluster system (HPCC) is also a possible solution in early stage of big data analytics. Sagiroglu and Sinanc [105] therefore compare the characteristics between HPCC and Hadoop. They then emphasized that HPCC system uses the multikey and multivariate indexes on distributed file system while Hadoop uses the

column-oriented database. In [17], Chen et al. give a brief introduction to the big data analytics of business intelligence (BI) from the perspective of evolution, applications, and emerging research topics. In their survey, Chen et al. explained that the revolution of business intelligence and analytics (BI&I) was from BI&I 1.0, BI&I 2.0, to BI&I 3.0 which are DBMS-based and structured content, web-based and unstructured content, and mobile and sensor based content, respectively.

Big data analysis algorithms

Mining algorithms for specific problem

Because the big data issues have appeared for nearly ten years, in [106], Fan and Bifet pointed out that the terms "big data" [107] and "big data mining" [108] were first presented in 1998, respectively. The big data and big data mining almost appearing at the same time explained that finding something from big data will be one of the major tasks in this research domain. Data mining algorithms for data analysis also play the vital role in the big data analysis, in terms of the computation cost, memory requirement, and accuracy of the end results. In this section, we will give a brief discussion from the perspective of analysis and search algorithms to explain its importance for big data analytics.

Clustering algorithms In the big data age, traditional clustering algorithms will become even more limited than before because they typically require that all the data be in the same format and be loaded into the same machine so as to find some useful things from the whole data. Although the problem [64] of analyzing large-scale and high-dimensional dataset has attracted many researchers from various disciplines in the last century, and several solutions [2, 109] have been presented presented in recent years, the characteristics of big data still brought up several new challenges for the data clustering issues. Among them, how to reduce the data complexity is one of the important issues for big data clustering. In [110], Shirkhorshidi et al. divided the big data clustering into two categories: single-machine clustering (i.e., sampling and dimension reduction solutions), and multiple-machine clustering (parallel and MapReduce solutions). This means that traditional reduction solutions can also be used in the big data age because the complexity and memory space needed for the process of data analysis will be decreased by using sampling and dimension reduction methods. More precisely, sampling can be regarded as reducing the "amount of data" entered into a data analyzing process while dimension reduction can be regarded as "downsizing the whole dataset" because irrelevant dimensions will be discarded before the data analyzing process is carried out.

CloudVista [111] is a representative solution for clustering big data which used cloud computing to perform the clustering process in parallel. BIRCH [44] and sampling method were used in CloudVista to show that it is able to handle large-scale data, e.g., 25 million census records. Using GPU to enhance the performance of a clustering algorithm is another promising solution for big data mining. The multiple species flocking (MSF) [112] was applied to the CUDA platform from NVIDIA to reduce the computation time of clustering algorithm in [113]. The simulation results show that the speedup factor can be increased from 30 up to 60 by using GPU for data clustering. Since most traditional clustering algorithms (e.g, k-means) require a computation that is centralized, how to make them capable of handling big data clustering problems is the major

concern of Feldman et al. [114] who use a tree construction for generating the coresets in parallel which is called the "merge-and-reduce" approach. Moreover, Feldman et al. pointed out that by using this solution for clustering, the update time per datum and memory of the traditional clustering algorithms can be significantly reduced.

Classification algorithms Similar to the clustering algorithm for big data mining, several studies also attempted to modify the traditional classification algorithms to make them work on a parallel computing environment or to develop new classification algorithms which work naturally on a parallel computing environment. In [115], the design of classification algorithm took into account the input data that are gathered by distributed data sources and they will be processed by a heterogeneous set of learners.[5] In this study, Tekin et al. presented a novel classification algorithm called "classify or send for classification" (CoS). They assumed that each learner can be used to process the input data in two different ways in a distributed data classification system. One is to perform a classification function by itself while the other is to forward the input data to another learner to have them labeled. The information will be exchanged between different learners. In brief, this kind of solutions can be regarded as a cooperative learning to improve the accuracy in solving the big data classification problem. An interesting solution uses the quantum computing to reduce the memory space and computing cost of a classification algorithm. For example, in [116], Rebentrost et al. presented a quantum-based support vector machine for big data classification and argued that the classification algorithm they proposed can be implemented with a time complexity $O(\log NM)$ where N is the number of dimensions and M is the number of training data. There are bright prospects for big data mining by using quantum-based search algorithm when the hardware of quantum computing has become mature.

Frequent pattern mining algorithms Most of the researches on frequent pattern mining (i.e., association rules and sequential pattern mining) were focused on handling large-scale dataset at the very beginning because some early approaches of them were attempted to analyze the data from the transaction data of large shopping mall. Because the number of transactions usually is more than "tens of thousands", the issues about how to handle the large scale data were studied for several years, such as FP-tree [32] using the tree structure to include the frequent patterns to further reduce the computation time of association rule mining. In addition to the traditional frequent pattern mining algorithms, of course, parallel computing and cloud computing technologies have also attracted researchers in this research domain. Among them, the map-reduce solution was used for the studies [117–119] to enhance the performance of the frequent pattern mining algorithm. By using the map-reduce model for frequent pattern mining algorithm, it can be easily expected that its application to "cloud platform" [120, 121] will definitely become a popular trend in the forthcoming future. The study of [119] no only used the map-reduce model, it also allowed users to express their specific interest constraints in the process of frequent pattern mining. The performance of these methods by using map-reduce model for big data analysis is, no doubt, better than the traditional frequent pattern mining algorithms running on a single machine.

[5] The learner typically represented the classification function which will create the classifier to help us classify the unknown input data.

Machine learning for big data mining

The potential of machine learning for data analytics can be easily found in the early literature [22, 49]. Different from the data mining algorithm design for specific problems, machine learning algorithms can be used for different mining and analysis problems because they are typically employed as the "search" algorithm of the required solution. Since most machine learning algorithms can be used to find an approximate solution for the optimization problem, they can be employed for most data analysis problems if the data analysis problems can be formulated as an optimization problem. For example, genetic algorithm, one of the machine learning algorithms, can not only be used to solve the clustering problem [25], it can also be used to solve the frequent pattern mining problem [33]. The potential of machine learning is not merely for solving different mining problems in data analysis operator of KDD; it also has the potential of enhancing the performance of the other parts of KDD, such as feature reduction for the input operators [72].

A recent study [68] shows that some traditional mining algorithms, statistical methods, preprocessing solutions, and even the GUI's have been applied to several representative tools and platforms for big data analytics. The results show clearly that machine learning algorithms will be one of the essential parts of big data analytics. One of the problems in using current machine learning methods for big data analytics is similar to those of most traditional data mining algorithms which are designed for sequential or centralized computing. However, one of the most possible solutions is to make them work for parallel computing. Fortunately, some of the machine learning algorithms (e.g., population-based algorithms) can essentially be used for parallel computing, which have been demonstrated for several years, such as parallel computing version of genetic algorithm [122]. Different from the traditional GA, as shown in Fig. 9a, the population of island model genetic algorithm, one of the parallel GA's, can be divided into several sub-populations, as shown in Fig. 9b. This means that the sub-populations can be assigned to different threads or computer nodes for parallel computing, by a simple modification of the GA.

For this reason, in [123], Kiran and Babu explained that the framework for distributed data mining algorithm still needs to aggregate the information from different computer nodes. As shown in Fig. 10, the common design of distributed data mining algorithm is as follows: each mining algorithm will be performed on a computer node (worker) which has its locally coherent data, but not the whole data. To construct a globally meaningful knowledge after each mining algorithm finds its local model, the local model from each computer node has to be aggregated and integrated into a final model to represent the complete knowledge. Kiran and Babu [123] also pointed out that the communication will be the bottleneck when using this kind of distributed computing framework.

Bu et al. [124] found some research issues when trying to apply machine learning algorithms to parallel computing platforms. For instance, the early version of map-reduce framework does not support "iteration" (i.e., recursion). But the good news is that some recent works [87, 125] have paid close attention to this problem and tried to fix it. Similar to the solutions for enhancing the performance of the traditional data mining algorithms, one of the possible solutions to enhancing the performance of a machine learning algorithm is to use CUDA, i.e., a GPU, to reduce the computing time of data

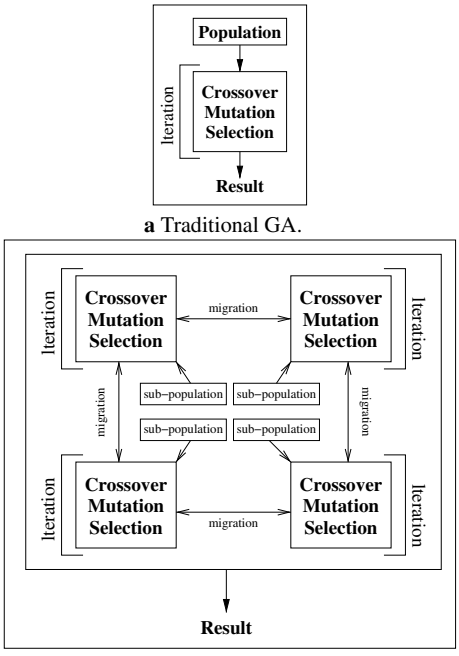


Fig. 9 The comparison between basic idea of traditional GA (TGA) and parallel genetic algorithm (PGA)

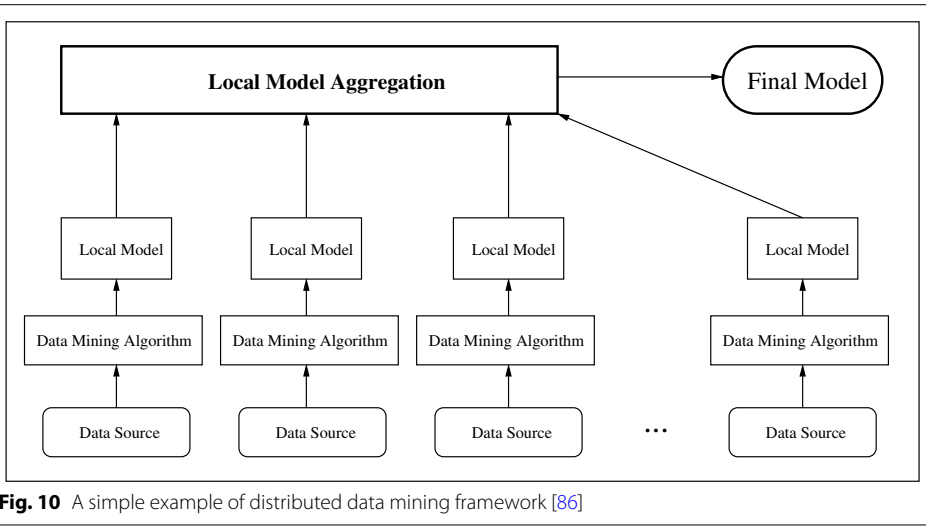


Fig. 10 A simple example of distributed data mining framework [86]

analysis. Hasan et al. [126] used CUDA to implement the self-organizing map (SOM) and multiple back-propagation (MBP) for the classification problem. The simulation results show that using GPU is faster than using CPU. More precisely, SOM running on a GPU is three times faster than SOM running on a CPU, and MPB running on a GPU is twenty-seven times faster than MPB running on a. Another study [127] attempted to apply the ant-based algorithm to grid computing platform. Since the proposed mining algorithm is extended by the ant clustering algorithm of Deneubourg et al. [128],[6]

[6] The basic idea of [128] is that each ant will pick up and drop data items in terms of the similarity of its local neighbors.

Ku-Mahamud modified the ant behavior of this ant clustering algorithm for big data clustering. That is, each ant will be randomly placed on the grid. This means that the ant clustering algorithm then can be used on a parallel computing environment.

The trends of machine learning studies for big data analytics can be divided into two-fold: one attempts to make machine learning algorithms run on parallel platforms, such as Radoop [129], Mahout [87], and PIMRU [124]; the other is to redesign the machine learning algorithms to make them suitable for parallel computing or to parallel computing environment, such as neural network algorithms for GPU [126] and ant-based algorithm for grid [127]. In summary, both of them make it possible to apply the machine learning algorithms to big data analytics although still many research issues need to be solved, such as the communication cost for different computer nodes [86] and the large computation cost most machine learning algorithms require [126].

Output the result of big data analysis

The benchmarks of PigMix [130], GridMix [131], TeraSort and GraySort [132], TPC-C, TPC-H, TPC-DS [133], and yahoo cloud serving benchmark (YCSB) [134] have been presented for evaluating the performance of the cloud computing and big data analytics systems. Ghazal et al. [135] presented another benchmark (called BigBench) to be used as an end-to-end big data benchmark which covers the characteristics of 3V of big data and uses the loading time, time for queries, time for procedural processing queries, and time for the remaining queries as the metrics. By using these benchmarks, the computation time is one of the intuitive metrics for evaluating the performance of different big data analytics platforms or algorithms. That is why Cheptsov [136] compered the high performance computing (HPC) and cloud system by using the measurement of computation time to understand their scalability for text file analysis. In addition to the computation time, the throughput (e.g., the number of operations per second) and read/write latency of operations are the other measurements of big data analytics [137]. In the study of [138], Zhao et al. believe that the maximum size of data and the maximum number of jobs are the two important metrics to understand the performance of the big data analytics platform. Another study described in [139] presented a systematic evaluation method which contains the data throughput, concurrency during map and reduce phases, response times, and the execution time of map and reduce. Moreover, most benchmarks for evaluating the performance of big data analytics typically can only provide the response time or the computation cost; however, the fact is that several factors need to be taken into account at the same time when building a big data analytics system. The hardware, bandwidth for data transmission, fault tolerance, cost, power consumption of these systems are all issues [70, 104] to be taken into account at the same time when building a big data analytics system. Several solutions available today are to install the big data analytics on a cloud computing system or a cluster system. Therefore, the measurements of fault tolerance, task execution, and cost of cloud computing systems can then be used to evaluate the performance of the corresponding factors of big data analytics.

How to present the analysis results to a user is another important work in the output part of big data analytics because if the user cannot easily understand the meaning of the results, the results will be entirely useless. Business intelligent and network monitoring are the two common approaches because their user interface plays the vital role of

making them workable. Zhang et al. [140] pointed out that the tasks of the visual analytics for commercial systems can be divided into four categories which are exploration, dashboards, reporting, and alerting. The study [141] showed that the interface for electroencephalography (EEG) interpretation is another noticeable research issue in big data analytics. The user interface for cloud system [142, 143] is the recent trend for big data analytics. This usually plays vital roles in big data analytics system, one of which is to simplify the explanation of the needed knowledge to the users while the other is to make it easier for the users to handle the data analytics system to work with their opinions. According to our observations, a flexible user interface is needed because although the big data analytics can help us to find some hidden information, the information found usually is not knowledge. This situation is just like the example we mentioned in "Output the result". The mining or statistical techniques can be employed to know the flu situation of each region, but data scientists sometimes need additional ways to display the information to find out the knowledge they need or to prove their assumption. Thus, the user interface can be adjusted by the user to display the knowledge that is needed urgently for big data analytics.

Summary of process of big data analytics

This discussion of big data analytics in this section was divided into input, analysis, and output for mapping the data analysis process of KDD. For the input (see also in "Big data input") and output (see also "Output the result of big data analysis") of big data, several methods and solutions proposed before the big data age (see also "Data input") can also be employed for big data analytics in most cases.

However, there still exist some new issues of the input and output that the data scientists need to confront. A representative example we mentioned in "Big data input" is that the bottleneck will not only on the sensor or input devices, it may also appear in other places of data analytics [71]. Although we can employ traditional compression and sampling technologies to deal with this problem, they can only mitigate the problems instead of solving the problems completely. Similar situations also exist in the output part. Although several measurements can be used to evaluate the performance of the frameworks, platforms, and even data mining algorithms, there still exist several new issues in the big data age, such as information fusion from different information sources or information accumulation from different times.

Several studies attempted to present an efficient or effective solution from the perspective of system (e.g., framework and platform) or algorithm level. A simple comparison of these big data analysis technologies from different perspectives is described in Table 3, to give a brief introduction to the current studies and trends of data analysis technologies for the big data. The "Perspective" column of this table explains that the study is focused on the framework or algorithm level; the "Description" column gives the further goal of the study; and the "Name" column is an abbreviated names of the methods or platform/framework. From the analysis framework perspective, this table shows that big data *framework*, *platform*, and *machine learning* are the current research trends in big data analytics system. For the mining algorithm perspective, the *clustering*, *classification*, and *frequent pattern mining* issues play the vital role of these researches because several data analysis problems can be mapped to these essential issues.

Table 3 The big data analysis frameworks and methods

$\mathcal{P}$	Name	References	Year	Description	$\mathcal{T}$
Analysis framework	DOT	[88]	2011	Add more computation resources via scale out solution	Framework
	GLADE	[89]	2011	Multi-level tree-based system architecture	
	Starfish	[92]	2012	Self-tuning analytics system	
	ODT-MDC	[96]	2012	Privacy issues	
	MRAM	[91]	2013	Mobile agent technologies	
	CBDMASP	[94]	2013	Statistical computation and data mining approaches	
	SODSS	[97]	2013	Decision support system issues	
	BDAF	[93]	2014	Data centric architecture	
	HACE	[95]	2014	Data mining approaches	
	Hadoop	[83]	2011	Parallel computing platform	Platform
	CUDA	[84]	2007	Parallel computing platform	
	Storm	[85]	2014	Parallel computing platform	
	Pregel	[125]	2010	Large-scale graph data analysis	
	MLPACK	[86]	2013	Scalable machine learning library	ML
	Mahout	[87]	2011	Machine-learning algorithms	
	MLAS	[124]	2012	Machine-learning algorithms	
	PIMRU	[124]	2012	Machine Learning algorithms	
	Radoop	[129]	2011	Data analytics, machine learning algorithms, and R statistical tool	
Mining algorithm	DBDC	[144]	2004	Parallel clustering	CLU
	PKM	[145]	2009	Map-reduce-based *k*-means clustering	
	CloudVista	[111]	2012	Cloud computing for clustering	
	MSFCUDA	[113]	2013	GPU for clustering	
	BDCAC	[127]	2013	Ant on grid computing environment for clustering	
	Corest	[114]	2013	Use a tree construction for generating the coresets in parallel for clustering	
	SOM-MBP	[126]	2013	Neural network with CGP for classification	CLA
	CoS	[115]	2013	Parallel computing for classification	
	SVMGA	[72]	2014	Using GA for reduce the number of dimensions	
	Quantum SVM	[116]	2014	Quantum computing for classification	
	DPSP	[121]	2010	Applied frequent pattern algorithm to cloud platform	FP
	DHTRIE	[120]	2011	Applied frequent pattern algorithm to cloud platform	
	SPC, FPC, and DPC	[117]	2012	Map-reduce model for frequent pattern mining	
	MFPSAM	[119]	2014	Concerned the specific interest constraints and applied map-reduce model	

$\mathcal{P}$ perspective, $\mathcal{T}$ taxonomy, *ML* machine learning, *CLU* clustering, *CLA* classification, *FP* frequent pattern

A promising trend that can be easily found from these successful examples is to use machine learning as the search algorithm (i.e., mining algorithm) for the data mining problems of big data analytics system. The machine learning-based methods are able to

make the mining algorithms and relevant platforms smarter or reduce the redundant computation costs. That parallel computing and cloud computing technologies have a strong impact on the big data analytics can also be recognized as follows: (1) most of the big data analytics frameworks and platforms are using Hadoop and Hadoop relevant technologies to design their solutions; and (2) most of the mining algorithms for big data analysis have been designed for parallel computing via software or hardware or designed for Map-Reduce-based platform.

From the results of recent studies of big data analytics, it is still at the early stage of Nolan's stages of growth model [146] which is similar to the situations for the research topics of cloud computing, internet of things, and smart grid. This is because several studies just attempted to apply the traditional solutions to the new problems/platforms/environments. For example, several studies [114, 145] used *k*-means as an example to analyze the big data, but not many studies applied the state-of-the-art data mining algorithms and machine learning algorithms to the analysis the big data. This explains that the performance of the big data analytics can be improved by data mining algorithms and metaheuristic algorithms presented in recent years [147]. The relevant technologies for compression, sampling, or even the platform presented in recent years may also be used to enhance the performance of the big data analytics system. As a result, although these research topics still have several open issues that need to be solved, these situations, on the contrary, also illustrate that everything is possible in these studies.

The open issues

Although the data analytics today may be inefficient for big data caused by the environment, devices, systems, and even problems that are quite different from traditional mining problems, because several characteristics of big data also exist in the traditional data analytics. Several open issues caused by the big data will be addressed as the platform/framework and data mining perspectives in this section to explain what dilemmas we may confront because of big data. Here are some of the open issues:

Platform and framework perspective

Input and output ratio of platform

A large number of reports and researches mentioned that we will enter the big data age in the near future. Some of them insinuated to us that these fruitful results of big data will lead us to a whole new world where "everything" is possible; therefore, the big data analytics will be an omniscient and omnipotent system. From the pragmatic perspective, the big data analytics is indeed useful and has many possibilities which can help us more accurately understand the so-called "things." However, the situation in most studies of big data analytics is that they argued that the results of big data are valuable, but the business models of most big data analytics are not clear. The fact is that assuming we have infinite computing resources for big data analytics is a thoroughly impracticable plan, the input and output ratio (e.g., return on investment) will need to be taken into account before an organization constructs the big data analytics center.

Communication between systems

Since most big data analytics systems will be designed for parallel computing, and they typically will work on other systems (e.g., cloud platform) or work with other systems (e.g., search engine or knowledge base), the communication between the big data analytics and other systems will strongly impact the performance of the whole process of KDD. The first research issue for the communication is that the communication cost will incur between systems of data analytics. How to reduce the communication cost will be the very first thing that the data scientists need to care. Another research issue for the communication is how the big data analytics communicates with other systems. The consistency of data between different systems, modules, and operators is also an important open issue on the communication between systems. Because the communication will appear more frequently between systems of big data analytics, how to reduce the cost of communication and how to make the communication between these systems as reliable as possible will be the two important open issues for big data analytics.

Bottlenecks on data analytics system

The bottlenecks will be appeared in different places of the data analytics for big data because the environments, systems, and input data have changed which are different from the traditional data analytics. The data deluge of big data will fill up the "input" system of data analytics, and it will also increase the computation load of the data "analysis" system. This situation is just like the torrent of water (i.e., data deluge) rushed down the mountain (i.e., data analytics), how to split it and how to avoid it flowing into a narrow place (e.g., the operator is not able to handle the input data) will be the most important things to avoid the bottlenecks in data analytics system. One of the current solutions to the avoidance of bottlenecks on a data analytics system is to add more computation resources while the other is to split the analysis works to different computation nodes. A complete consideration for the whole data analytics to avoid the bottlenecks of that kind of analytics system is still needed for big data.

Security issues

Since much more environment data and human behavior will be gathered to the big data analytics, how to protect them will also be an open issue because without a security way to handle the collected data, the big data analytics cannot be a reliable system. In spite of the security that we have to tighten for big data analytics before it can gather more data from everywhere, the fact is that until now, there are still not many studies focusing on the security issues of the big data analytics. According to our observation, the security issues of big data analytics can be divided into fourfold: input, data analysis, output, and communication with other systems. For the input, it can be regarded as the data gathering which is relevant to the sensor, the handheld devices, and even the devices of internet of things. One of the important security issues on the input part of big data analytics is to make sure that the sensors will not be compromised by the attacks. For the analysis and input, it can be regarded as the security problem of such a system. For communication with other system, the security problem is on the communications between big data analytics and other external systems. Because of these latent problems, security has become one of the open issues of big data analytics.

Data mining perspective

Data mining algorithm for map-reduce solution

As we mentioned in the previous sections, most of the traditional data mining algorithms are not designed for parallel computing; therefore, they are not particularly useful for the big data mining. Several recent studies have attempted to modify the traditional data mining algorithms to make them applicable to Hadoop-based platforms. As long as porting the data mining algorithms to Hadoop is inevitable, making the data mining algorithms work on a map-reduce architecture is the first very thing to do to apply traditional data mining methods to big data analytics. Unfortunately, not many studies attempted to make the data mining and soft computing algorithms work on Hadoop because several different backgrounds are needed to develop and design such algorithms. For instance, the researcher and his or her research group need to have the background in data mining and Hadoop so as to develop and design such algorithms. Another open issue is that most data mining algorithms are designed for centralized computing; that is, they can only work on all the data at the same time. Thus, how to make them work on a parallel computing system is also a difficult work. The good news is that some studies [145] have successfully applied the traditional data mining algorithms to the map-reduce architecture. These results imply that it is possible to do so. According to our observation, although the traditional mining or soft computing algorithms can be used to help us analyze the data in big data analytics, unfortunately, until now, not many studies are focused on it. As a consequence, it is an important open issue in big data analytics.

Noise, outliers, incomplete and inconsistent data

Although big data analytics is a new age for data analysis, because several solutions adopt classical ways to analyze the data on big data analytics, the open issues of traditional data mining algorithms also exist in these new systems. The open issues of noise, outliers, incomplete, and inconsistent data in traditional data mining algorithms will also appear in big data mining algorithms. More incomplete and inconsistent data will easily appear because the data are captured by or generated from different sensors and systems. The impact of noise, outliers, incomplete and inconsistent data will be enlarged for big data analytics. Therefore, how to mitigate the impact will be the open issues for big data analytics.

Bottlenecks on data mining algorithm

Most of the data mining algorithms in big data analytics will be designed for parallel computing. However, once data mining algorithms are designed or modified for parallel computing, it is the information exchange between different data mining procedures that may incur bottlenecks. One of them is the synchronization issue because different mining procedures will finish their jobs at different times even though they use the same mining algorithm to work on the same amount of data. Thus, some of the mining procedures will have to wait until the others finished their jobs. This situation may occur because the loading of different computer nodes may be different during the data mining process, or it may occur because the convergence speeds are different for the same data mining algorithm. The bottlenecks of data mining algorithms will become an open issue

for the big data analytics which explains that we need to take into account this issue when we develop and design a new data mining algorithm for big data analytics.

Privacy issues

The privacy concern typically will make most people uncomfortable, especially if systems cannot guarantee that their personal information will not be accessed by the other people and organizations. Different from the concern of the security, the privacy issue is about if it is possible for the system to restore or infer personal information from the results of big data analytics, even though the input data are anonymous. The privacy issue has become a very important issue because the data mining and other analysis technologies will be widely used in big data analytics, the private information may be exposed to the other people after the analysis process. For example, although all the gathered data for shop behavior are anonymous (e.g., buying a pistol), because the data can be easily collected by different devices and systems (e.g., location of the shop and age of the buyer), a data mining algorithm can easily infer who bought this pistol. More precisely, the data analytics is able to reduce the scope of the database because location of the shop and age of the buyer provide the information to help the system find out possible persons. For this reason, any sensitive information needs to be carefully protected and used. The anonymous, temporary identification, and encryption are the representative technologies for privacy of data analytics, but the critical factor is how to use, what to use, and why to use the collected data on big data analytics.

Conclusions

In this paper, we reviewed studies on the data analytics from the traditional data analysis to the recent big data analysis. From the system perspective, the KDD process is used as the framework for these studies and is summarized into three parts: input, analysis, and output. From the perspective of big data analytics framework and platform, the discussions are focused on the performance-oriented and results-oriented issues. From the perspective of data mining problem, this paper gives a brief introduction to the data and big data mining algorithms which consist of clustering, classification, and frequent patterns mining technologies. To better understand the changes brought about by the big data, this paper is focused on the data analysis of KDD from the platform/framework to data mining. The open issues on computation, quality of end result, security, and privacy are then discussed to explain which open issues we may face. Last but not least, to help the audience of the paper find *solutions* to welcome the new age of big data, the possible high impact research trends are given below:

- For the computation time, there is no doubt at all that parallel computing is one of the important future trends to make the data analytics work for big data, and consequently the technologies of cloud computing, Hadoop, and map-reduce will play the important roles for the big data analytics. To handle the computation resources of the cloud-based platform and to finish the task of data analysis as fast as possible, the scheduling method is another future trend.
- Using efficient methods to reduce the computation time of input, comparison, sampling, and a variety of reduction methods will play an important role in big data analyt-

ics. Because these methods typically do not consider parallel computing environment, how to make them work on parallel computing environment will be a future research trend. Similar to the input, the data mining algorithms also face the same situation that we mentioned in the previous section , how to make them work on parallel computing environment will be a very important research trend because there are abundant research results on traditional data mining algorithms.

- How to model the mining problem to find *something* from big data and how to display the knowledge we got from big data analytics will also be another two vital future trends because the results of these two researches will decide if the data analytics can practically work for real world approaches, not just a theoretical stuff.
- The methods of extracting information from external and relative knowledge resources to further reinforce the big data analytics, until now, are not very popular in big data analytics. But combining information from different resources to add the value of output knowledge is a common solution in the area of information retrieval, such as clustering search engine or document summarization. For this reason, information fusion will also be a future trend for improving the end results of big data analytics.
- Because the metaheuristic algorithms are capable of finding an approximate solution within a reasonable time, they have been widely used in solving the data mining problem in recent years. Until now, many state-of-the-art metaheuristic algorithms still have not been applied to big data analytics. In addition, compared to some early data mining algorithms, the performance of metaheuristic is no doubt superior in terms of the computation time and the quality of end result. From these observations, the application of metaheuristic algorithms to big data analytics will also be an important research topic.
- Because social network is part of the daily life of most people and because its data is also a kind of big data, how to analyze the data of a social network has become a promising research issue. Obviously, it can be used to predict the behavior of a user. After that, we can make applicable strategies for the user. For instance, a business intelligence system can use the analysis results to encourage particular customers to buy the goods they are interested.
- The security and privacy issues that accompany the work of data analysis are intuitive research topics which contain how to safely store the data, how to make sure the data communication is protected, and how to prevent someone from finding out the information about us. Many problems of data security and privacy are essentially the same as those of the traditional data analysis even if we are entering the big data age. Thus, how to protect the data will also appear in the research of big data analytics.

Abbreviations

PCA: principal components analysis; 3Vs: volume, velocity, and variety; IDC: International Data Corporation; KDD: knowledge discovery in databases; SVM: support vector machine; SSE: sum of squared errors; GLADE: generalized linear aggregates distributed engine; BDAF: big data architecture framework; CBDMASP: cloud-based big data mining & analyzing services platform; SODSS: service-oriented decision support system; HPCC: high performance computing cluster system; BI&I: business intelligence and analytics; DBMS: database management system; MSF: multiple species flocking; GA: genetic algorithm; SOM: self-organizing map; MBP: multiple back-propagation; YCSB: yahoo cloud serving benchmark; HPC: high performance computing; EEG: electroencephalography.

Authors' contributions

CWT contributed to the paper review and drafted the first version of the manuscript. CFL contributed to the paper collection and manuscript organization. HCC and AVV double checked the manuscript and provided several advanced ideas for this manuscript. All authors read and approved the final manuscript.

Author details
[1] Department of Computer Science and Information Engineering, National Ilan University, Yilan, Taiwan. [2] Institute of Computer Science and Information Engineering, National Chung Cheng University, Chia-Yi, Taiwan. [3] Information Engineering College, Yangzhou University, Yangzhou, Jiangsu, China. [4] School of Information Science and Engineering, Fujian University of Technology, Fuzhou, Fujian, China. [5] Department of Computer Science, Electrical and Space Engineering, Luleå University of Technology, SE-931 87 Skellefteå, Sweden.

Acknowledgements
The authors would like to thank the anonymous reviewers for their valuable comments and suggestions on the paper. This work was supported in part by the Ministry of Science and Technology of Taiwan, R.O.C., under Contracts MOST103-2221-E-197-034, MOST104-2221-E-197-005, and MOST104-2221-E-197-014.

Compliance with ethical guidelines

Competing interests
The authors declare that they have no competing interests.

Received: 14 May 2015 Accepted: 2 September 2015
Published online: 01 October 2015

References

1. Lyman P, Varian H. How much information 2003? Tech. Rep, 2004. [Online]. Available: http://www2.sims.berkeley.edu/research/projects/how-much-info-2003/printable_report.pdf.
2. Xu R, Wunsch D. Clustering. Hoboken: Wiley-IEEE Press; 2009.
3. Ding C, He X. K-means clustering via principal component analysis. In: Proceedings of the Twenty-first International Conference on Machine Learning, 2004, pp 1–9.
4. Kollios G, Gunopulos D, Koudas N, Berchtold S. Efficient biased sampling for approximate clustering and outlier detection in large data sets. IEEE Trans Knowl Data Eng. 2003;15(5):1170–87.
5. Fisher D, DeLine R, Czerwinski M, Drucker S. Interactions with big data analytics. Interactions. 2012;19(3):50–9.
6. Laney D. 3D data management: controlling data volume, velocity, and variety, META Group, Tech. Rep. 2001. [Online]. Available: http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf.
7. van Rijmenam M. Why the 3v's are not sufficient to describe big data, BigData Startups, Tech. Rep. 2013. [Online]. Available: http://www.bigdata-startups.com/3vs-sufficient-describe-big-data/.
8. Borne K. Top 10 big data challenges a serious look at 10 big data v's, Tech. Rep. 2014. [Online]. Available: https://www.mapr.com/blog/top-10-big-data-challenges-look-10-big-data-v.
9. Press G. $16.1 billion big data market: 2014 predictions from IDC and IIA, Forbes, Tech. Rep. 2013. [Online]. Available: http://www.forbes.com/sites/gilpress/2013/12/12/16-1-billion-big-data-market-2014-predictions-from-idc-and-iia/.
10. Big data and analytics—an IDC four pillar research area, IDC, Tech. Rep. 2013. [Online]. Available: http://www.idc.com/prodserv/FourPillars/bigData/index.jsp.
11. Taft DK. Big data market to reach $46.34 billion by 2018, EWEEK, Tech. Rep. 2013. [Online]. Available: http://www.eweek.com/database/big-data-market-to-reach-46.34-billion-by-2018.html.
12. Research A. Big data spending to reach $114 billion in 2018; look for machine learning to drive analytics, ABI Research, Tech. Rep. 2013. [Online]. Available: https://www.abiresearch.com/press/big-data-spending-to-reach-114-billion-in-2018-loo.
13. Furrier J. Big data market $50 billion by 2017—HP vertica comes out #1—according to wikibon research, SiliconANGLE, Tech. Rep. 2012. [Online]. Available: http://siliconangle.com/blog/2012/02/15/big-data-market-15-billion-by-2017-hp-vertica-comes-out-1-according-to-wikibon-research/.
14. Kelly J, Vellante D, Floyer D. Big data market size and vendor revenues, Wikibon, Tech. Rep. 2014. [Online]. Available: http://wikibon.org/wiki/v/Big_Data_Market_Size_and_Vendor_Revenues.
15. Kelly J, Floyer D, Vellante D, Miniman S. Big data vendor revenue and market forecast 2012-2017, Wikibon, Tech. Rep. 2014. [Online]. Available: http://wikibon.org/wiki/v/Big_Data_Vendor_Revenue_and_Market_Forecast_2012-2017.
16. Mayer-Schonberger V, Cukier K. Big data: a revolution that will transform how we live, work, and think. Boston: Houghton Mifflin Harcourt; 2013.
17. Chen H, Chiang RHL, Storey VC. Business intelligence and analytics: from big data to big impact. MIS Quart. 2012;36(4):1165–88.
18. Kitchin R. The real-time city? big data and smart urbanism. Geo J. 2014;79(1):1–14.
19. Fayyad UM, Piatetsky-Shapiro G, Smyth P. From data mining to knowledge discovery in databases. AI Mag. 1996;17(3):37–54.
20. Han J. Data mining: concepts and techniques. San Francisco: Morgan Kaufmann Publishers Inc.; 2005.
21. Agrawal R, Imieliński T, Swami A. Mining association rules between sets of items in large databases. Proc ACM SIGMOD Int Conf Manag Data. 1993;22(2):207–16.
22. Witten IH, Frank E. Data mining: practical machine learning tools and techniques. San Francisco: Morgan Kaufmann Publishers Inc.; 2005.
23. Abbass H, Newton C, Sarker R. Data mining: a heuristic approach. Hershey: IGI Global; 2002.
24. Cannataro M, Congiusta A, Pugliese A, Talia D, Trunfio P. Distributed data mining on grids: services, tools, and applications. IEEE Trans Syst Man Cyber Part B Cyber. 2004;34(6):2451–65.

25. Krishna K, Murty MN. Genetic *k*-means algorithm. IEEE Trans Syst Man Cyber Part B Cyber. 1999;29(3):433–9.
26. Tsai C-W, Lai C-F, Chiang M-C, Yang L. Data mining for internet of things: a survey. IEEE Commun Surveys Tutor. 2014;16(1):77–97.
27. Jain AK, Murty MN, Flynn PJ. Data clustering: a review. ACM Comp Surveys. 1999;31(3):264–323.
28. McQueen JB. Some methods of classification and analysis of multivariate observations. In: Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability, 1967. pp 281–297.
29. Safavian S, Landgrebe D. A survey of decision tree classifier methodology. IEEE Trans Syst Man Cyber. 1991;21(3):660–74.
30. McCallum A, Nigam K. A comparison of event models for naive bayes text classification. In: Proceedings of the National Conference on Artificial Intelligence, 1998. pp. 41–48.
31. Boser BE, Guyon IM, Vapnik VN. A training algorithm for optimal margin classifiers. In: Proceedings of the annual workshop on Computational learning theory, 1992. pp. 144–152.
32. Han J, Pei J, Yin Y. Mining frequent patterns without candidate generation. In : Proceedings of the ACM SIGMOD International Conference on Management of Data, 2000. pp. 1–12.
33. Kaya M, Alhajj R. Genetic algorithm based framework for mining fuzzy association rules. Fuzzy Sets Syst. 2005;152(3):587–601.
34. Srikant R, Agrawal R. Mining sequential patterns: generalizations and performance improvements. In: Proceedings of the International Conference on Extending Database Technology: Advances in Database Technology, 1996. pp 3–17.
35. Zaki MJ. Spade: an efficient algorithm for mining frequent sequences. Mach Learn. 2001;42(1–2):31–60.
36. Baeza-Yates RA, Ribeiro-Neto B. Modern Information Retrieval. Boston: Addison-Wesley Longman Publishing Co., Inc; 1999.
37. Liu B. Web data mining: exploring hyperlinks, contents, and usage data. Berlin, Heidelberg: Springer-Verlag; 2007.
38. d'Aquin M, Jay N. Interpreting data mining results with linked data for learning analytics: motivation, case study and directions. In: Proceedings of the International Conference on Learning Analytics and Knowledge, pp 155–164.
39. Shneiderman B. The eyes have it: a task by data type taxonomy for information visualizations. In: Proceedings of the IEEE Symposium on Visual Languages, 1996, pp 336–343.
40. Mani I, Bloedorn E. Multi-document summarization by graph search and matching. In: Proceedings of the National Conference on Artificial Intelligence and Ninth Conference on Innovative Applications of Artificial Intelligence, 1997, pp 622–628.
41. Kopanakis I, Pelekis N, Karanikas H, Mavroudkis T. Visual techniques for the interpretation of data mining outcomes. In: Proceedings of the Panhellenic Conference on Advances in Informatics, 2005. pp 25–35.
42. Elkan C. Using the triangle inequality to accelerate k-means. In: Proceedings of the International Conference on Machine Learning, 2003, pp 147–153.
43. Catanzaro B, Sundaram N, Keutzer K. Fast support vector machine training and classification on graphics processors. In: Proceedings of the International Conference on Machine Learning, 2008. pp 104–111.
44. Zhang T, Ramakrishnan R, Livny M. BIRCH: an efficient data clustering method for very large databases. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, 1996. pp 103–114.
45. Ester M, Kriegel HP, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, 1996. pp 226–231.
46. Ester M, Kriegel HP, Sander J, Wimmer M, Xu X. Incremental clustering for mining in a data warehousing environment. In: Proceedings of the International Conference on Very Large Data Bases, 1998. pp 323–333.
47. Ordonez C, Omiecinski E. Efficient disk-based k-means clustering for relational databases. IEEE Trans Knowl Data Eng. 2004;16(8):909–21.
48. Kogan J. Introduction to clustering large and high-dimensional data. Cambridge: Cambridge Univ Press; 2007.
49. Mitra S, Pal S, Mitra P. Data mining in soft computing framework: a survey. IEEE Trans Neural Netw. 2002;13(1):3–14.
50. Mehta M, Agrawal R, Rissanen J. SLIQ: a fast scalable classifier for data mining. In: Proceedings of the 5th International Conference on Extending Database Technology: Advances in Database Technology. 1996. pp 18–32.
51. Micó L, Oncina J, Carrasco RC. A fast branch and bound nearest neighbour classifier in metric spaces. Pattern Recogn Lett. 1996;17(7):731–9.
52. Djouadi A, Bouktache E. A fast algorithm for the nearest-neighbor classifier. IEEE Trans Pattern Anal Mach Intel. 1997;19(3):277–82.
53. Ververidis D, Kotropoulos C. Fast and accurate sequential floating forward feature selection with the bayes classifier applied to speech emotion recognition. Signal Process. 2008;88(12):2956–70.
54. Pei J, Han J, Mao R. CLOSET: an efficient algorithm for mining frequent closed itemsets. In: Proceedings of the ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, 2000. pp 21–30.
55. Zaki MJ, Hsiao C-J. Efficient algorithms for mining closed itemsets and their lattice structure. IEEE Trans Knowl Data Eng. 2005;17(4):462–78.
56. Burdick D, Calimlim M, Gehrke J. MAFIA: a maximal frequent itemset algorithm for transactional databases. In: Proceedings of the International Conference on Data Engineering, 2001. pp 443–452.
57. Chen B, Haas P, Scheuermann P. A new two-phase sampling based algorithm for discovering association rules. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2002. pp 462–468.
58. Zaki MJ. SPADE: an efficient algorithm for mining frequent sequences. Mach Learn. 2001;42(1–2):31–60.
59. Yan X, Han J, Afshar R. CloSpan: mining closed sequential patterns in large datasets. In: Proceedings of the SIAM International Conference on Data Mining, 2003. pp 166–177.
60. Pei J, Han J, Asl MB, Pinto H, Chen Q, Dayal U, Hsu MC. PrefixSpan mining sequential patterns efficiently by prefix projected pattern growth. In: Proceedings of the International Conference on Data Engineering, 2001. pp 215–226.

61. Ayres J, Flannick J, Gehrke J, Yiu T. Sequential PAttern Mining using a bitmap representation. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2002. pp 429–435.
62. Masseglia F, Poncelet P, Teisseire M. Incremental mining of sequential patterns in large databases. Data Knowl Eng. 2003;46(1):97–121.
63. Xu R, Wunsch-II DC. Survey of clustering algorithms. IEEE Trans Neural Netw. 2005;16(3):645–78.
64. Chiang M-C, Tsai C-W, Yang C-S. A time-efficient pattern reduction algorithm for k-means clustering. Inform Sci. 2011;181(4):716–31.
65. Bradley PS, Fayyad UM. Refining initial points for k-means clustering. In: Proceedings of the International Conference on Machine Learning, 1998. pp 91–99.
66. Laskov P, Gehl C, Krüger S, Müller K-R. Incremental support vector learning: analysis, implementation and applications. J Mach Learn Res. 2006;7:1909–36.
67. Russom P. Big data analytics. TDWI: Tech. Rep ; 2011.
68. Ma C, Zhang HH, Wang X. Machine learning for big data analytics in plants. Trends Plant Sci. 2014;19(12):798–808.
69. Boyd D, Crawford K. Critical questions for big data. Inform Commun Soc. 2012;15(5):662–79.
70. Katal A, Wazid M, Goudar R. Big data: issues, challenges, tools and good practices. In: Proceedings of the International Conference on Contemporary Computing, 2013. pp 404–409.
71. Baraniuk RG. More is less: signal processing and the data deluge. Science. 2011;331(6018):717–9.
72. Lee J, Hong S, Lee JH. An efficient prediction for heavy rain from big weather data using genetic algorithm. In: Proceedings of the International Conference on Ubiquitous Information Management and Communication, 2014. pp 25:1–25:7.
73. Famili A, Shen W-M, Weber R, Simoudis E. Data preprocessing and intelligent data analysis. Intel Data Anal. 1997;1(1–4):3–23.
74. Zhang H. A novel data preprocessing solution for large scale digital forensics investigation on big data, Master's thesis, Norway, 2013.
75. Ham YJ, Lee H-W. International journal of advances in soft computing and its applications. Calc Paralleles Reseaux et Syst Repar. 2014;6(1):1–18.
76. Cormode G, Duffield N. Sampling for big data: a tutorial. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2014. pp 1975–1975.
77. Satyanarayana A. Intelligent sampling for big data using bootstrap sampling and chebyshev inequality. In: Proceedings of the IEEE Canadian Conference on Electrical and Computer Engineering, 2014. pp 1–6.
78. Jun SW, Fleming K, Adler M, Emer JS. Zip-io: architecture for application-specific compression of big data. In: Proceedings of the International Conference on Field-Programmable Technology, 2012, pp 343–351.
79. Zou H, Yu Y, Tang W, Chen HM. Improving I/O performance with adaptive data compression for big data applications. In: Proceedings of the International Parallel and Distributed Processing Symposium Workshops, 2014. pp 1228–1237.
80. Yang C, Zhang X, Zhong C, Liu C, Pei J, Ramamohanarao K, Chen J. A spatiotemporal compression based approach for efficient big data processing on cloud. J Comp Syst Sci. 2014;80(8):1563–83.
81. Xue Z, Shen G, Li J, Xu Q, Zhang Y, Shao J. Compression-aware I/O performance analysis for big data clustering. In: Proceedings of the International Workshop on Big Data, Streams and Heterogeneous Source Mining: Algorithms, Systems, Programming Models and Applications, 2012. pp 45–52.
82. Pospiech M, Felden C. Big data—a state-of-the-art. In: Proceedings of the Americas Conference on Information Systems, 2012, pp 1–23. [Online]. Available: http://aisel.aisnet.org/amcis2012/proceedings/DecisionSupport/22.
83. Apache Hadoop, February 2, 2015. [Online]. Available: http://hadoop.apache.org.
84. Cuda, February 2, 2015. [Online]. Available: URL: http://www.nvidia.com/object/cuda_home_new.html.
85. Apache Storm, February 2, 2015. [Online]. Available: URL: http://storm.apache.org/.
86. Curtin RR, Cline JR, Slagle NP, March WB, Ram P, Mehta NA, Gray AG. MLPACK: a scalable C++ machine learning library. J Mach Learn Res. 2013;14:801–5.
87. Apache Mahout, February 2, 2015. [Online]. Available: http://mahout.apache.org/.
88. Huai Y, Lee R, Zhang S, Xia CH, Zhang X. DOT: a matrix model for analyzing, optimizing and deploying software for big data analytics in distributed systems. In: Proceedings of the ACM Symposium on Cloud Computing, 2011. pp 4:1–4:14.
89. Rusu F, Dobra A. GLADE: a scalable framework for efficient analytics. In: Proceedings of LADIS Workshop held in conjunction with VLDB, 2012. pp 1–6.
90. Cheng Y, Qin C, Rusu F. GLADE: big data analytics made easy. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, 2012. pp 697–700.
91. Essa YM, Attiya G, El-Sayed A. Mobile agent based new framework for improving big data analysis. In: Proceedings of the International Conference on Cloud Computing and Big Data. 2013, pp 381–386.
92. Wonner J, Grosjean J, Capobianco A, Bechmann D Starfish: a selection technique for dense virtual environments. In: Proceedings of the ACM Symposium on Virtual Reality Software and Technology, 2012. pp 101–104.
93. Demchenko Y, de Laat C, Membrey P. Defining architecture components of the big data ecosystem. In: Proceedings of the International Conference on Collaboration Technologies and Systems, 2014. pp 104–112.
94. Ye F, Wang ZJ, Zhou FC, Wang YP, Zhou YC. Cloud-based big data mining and analyzing services platform integrating r. In: Proceedings of the International Conference on Advanced Cloud and Big Data, 2013. pp 147–151.
95. Wu X, Zhu X, Wu G-Q, Ding W. Data mining with big data. IEEE Trans Knowl Data Eng. 2014;26(1):97–107.
96. Laurila JK, Gatica-Perez D, Aad I, Blom J, Bornet O, Do T, Dousse O, Eberle J, Miettinen M. The mobile data challenge: big data for mobile computing research. In: Proceedings of the Mobile Data Challenge by Nokia Workshop, 2012. pp 1–8.
97. Demirkan H, Delen D. Leveraging the capabilities of service-oriented decision support systems: putting analytics and big data in cloud. Decision Support Syst. 2013;55(1):412–21.
98. Talia D. Clouds for scalable big data analytics. Computer. 2013;46(5):98–101.

99. Lu R, Zhu H, Liu X, Liu JK, Shao J. Toward efficient and privacy-preserving computing in big data era. IEEE Netw. 2014;28(4):46–50.
100. Cuzzocrea A, Song IY, Davis KC. Analytics over large-scale multidimensional data: The big data revolution!. In: Proceedings of the ACM International Workshop on Data Warehousing and OLAP, 2011. pp 101–104.
101. Zhang J, Huang ML. 5Ws model for big data analysis and visualization. In: Proceedings of the International Conference on Computational Science and Engineering, 2013. pp 1021–1028.
102. Chandarana P, Vijayalakshmi M. Big data analytics frameworks. In: Proceedings of the International Conference on Circuits, Systems, Communication and Information Technology Applications, 2014. pp 430–434.
103. Apache Drill February 2, 2015. [Online]. Available: URL: http://drill.apache.org/.
104. Hu H, Wen Y, Chua T-S, Li X. Toward scalable systems for big data analytics: a technology tutorial. IEEE Access. 2014;2:652–87.
105. Sagiroglu S, Sinanc D, Big data: a review. In: Proceedings of the International Conference on Collaboration Technologies and Systems, 2013. pp 42–47.
106. Fan W, Bifet A. Mining big data: current status, and forecast to the future. ACM SIGKDD Explor Newslett. 2013;14(2):1–5.
107. Diebold FX. On the origin(s) and development of the term "big data", Penn Institute for Economic Research, Department of Economics, University of Pennsylvania, Tech. Rep. 2012. [Online]. Available: http://economics.sas.upenn.edu/sites/economics.sas.upenn.edu/files/12-037.pdf.
108. Weiss SM, Indurkhya N. Predictive data mining: a practical guide. San Francisco: Morgan Kaufmann Publishers Inc.; 1998.
109. Fahad A, Alshatri N, Tari Z, Alamri A, Khalil I, Zomaya A, Foufou S, Bouras A. A survey of clustering algorithms for big data: taxonomy and empirical analysis. IEEE Trans Emerg Topics Comp. 2014;2(3):267–79.
110. Shirkhorshidi AS, Aghabozorgi SR, Teh YW, Herawan T. Big data clustering: a review. In: Proceedings of the International Conference on Computational Science and Its Applications, 2014. pp 707–720.
111. Xu H, Li Z, Guo S, Chen K. Cloudvista: interactive and economical visual cluster analysis for big data in the cloud. Proc VLDB Endowment. 2012;5(12):1886–9.
112. Cui X, Gao J, Potok TE. A flocking based algorithm for document clustering analysis. J Syst Archit. 2006;52(89):505–15.
113. Cui X, Charles JS, Potok T. GPU enhanced parallel computing for large scale data clustering. Future Gener Comp Syst. 2013;29(7):1736–41.
114. Feldman D, Schmidt M, Sohler C. Turning big data into tiny data: Constant-size coresets for k-means, pca and projective clustering. In: Proceedings of the ACM-SIAM Symposium on Discrete Algorithms, 2013. pp 1434–1453.
115. Tekin C, van der Schaar M. Distributed online big data classification using context information. In: Proceedings of the Allerton Conference on Communication, Control, and Computing, 2013. pp 1435–1442.
116. Rebentrost P, Mohseni M, Lloyd S. Quantum support vector machine for big feature and big data classification. *CoRR*, vol. abs/1307.0471, 2014. [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr1307.html#RebentrostML13.
117. Lin MY, Lee PY, Hsueh SC. Apriori-based frequent itemset mining algorithms on mapreduce. In: Proceedings of the International Conference on Ubiquitous Information Management and Communication, 2012. pp 76:1–76:8.
118. Riondato M, DeBrabant JA, Fonseca R, Upfal E. PARMA: a parallel randomized algorithm for approximate association rules mining in mapreduce. In: Proceedings of the ACM International Conference on Information and Knowledge Management, 2012. pp 85–94.
119. Leung CS, MacKinnon R, Jiang F. Reducing the search space for big data mining for interesting patterns from uncertain data. In: Proceedings of the International Congress on Big Data, 2014. pp 315–322.
120. Yang L, Shi Z, Xu L, Liang F, Kirsh I. DH-TRIE frequent pattern mining on hadoop using JPA. In: Proceedings of the International Conference on Granular Computing, 2011. pp 875–878.
121. Huang JW, Lin SC, Chen MS. DPSP: Distributed progressive sequential pattern mining on the cloud. In: Proceedings of the Advances in Knowledge Discovery and Data Mining, vol. 6119, 2010, pp 27–34.
122. Paz CE. A survey of parallel genetic algorithms. Calc Paralleles Reseaux et Syst Repar. 1998;10(2):141–71.
123. kranthi Kiran B, Babu AV. A comparative study of issues in big data clustering algorithm with constraint based genetic algorithm for associative clustering. Int J Innov Res Comp Commun Eng 2014; 2(8): 5423–5432.
124. Bu Y, Borkar VR, Carey MJ, Rosen J, Polyzotis N, Condie T, Weimer M, Ramakrishnan R. Scaling datalog for machine learning on big data, *CoRR*, vol. abs/1203.0160, 2012. [Online]. Available: http://dblp.uni-trier.de/db/journals/corr/corr1203.html#abs-1203-0160.
125. Malewicz G, Austern MH, Bik AJ, Dehnert JC, Horn I, Leiser N, Czajkowski G. Pregel: A system for large-scale graph processing. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, 2010. pp 135–146.
126. Hasan S, Shamsuddin S, Lopes N. Soft computing methods for big data problems. In: Proceedings of the Symposium on GPU Computing and Applications, 2013. pp 235–247.
127. Ku-Mahamud KR. Big data clustering using grid computing and ant-based algorithm. In: Proceedings of the International Conference on Computing and Informatics, 2013. pp 6–14.
128. Deneubourg JL, Goss S, Franks N, Sendova-Franks A, Detrain C, Chrétien L. The dynamics of collective sorting robot-like ants and ant-like robots. In: Proceedings of the International Conference on Simulation of Adaptive Behavior on From Animals to Animats, 1990. pp 356–363.
129. Radoop [Online]. https://rapidminer.com/products/radoop/. Accessed 2 Feb 2015.
130. PigMix [Online]. https://cwiki.apache.org/confluence/display/PIG/PigMix. Accessed 2 Feb 2015.
131. GridMix [Online]. http://hadoop.apache.org/docs/r1.2.1/gridmix.html. Accessed 2 Feb 2015.
132. TeraSoft [Online]. http://sortbenchmark.org/. Accessed 2 Feb 2015.
133. TPC, transaction processing performance council [Online]. http://www.tpc.org/. Accessed 2 Feb 2015.
134. Cooper BF, Silberstein A, Tam E, Ramakrishnan R, Sears R. Benchmarking cloud serving systems with ycsb. In: Proceedings of the ACM Symposium on Cloud Computing, 2010. pp 143–154.

135. Ghazal A, Rabl T, Hu M, Raab F, Poess M, Crolotte A, Jacobsen HA. BigBench: Towards an industry standard benchmark for big data analytics. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, 2013. pp 1197–1208.
136. Cheptsov A. Hpc in big data age: An evaluation report for java-based data-intensive applications implemented with hadoop and openmpi. In: Proceedings of the European MPI Users' Group Meeting, 2014. pp 175:175–175:180.
137. Yuan LY, Wu L, You JH, Chi Y. Rubato db: A highly scalable staged grid database system for oltp and big data applications. In: Proceedings of the ACM International Conference on Conference on Information and Knowledge Management, 2014. pp 1–10.
138. Zhao JM, Wang WS, Liu X, Chen YF. Big data benchmark - big DS. In: Proceedings of the Advancing Big Data Benchmarks, 2014, pp. 49–57.
139. Saletore V, Krishnan K, Viswanathan V, Tolentino M. HcBench: Methodology, development, and full-system characterization of a customer usage representative big data/hadoop benchmark. In: Advancing Big Data Benchmarks, 2014. pp 73–93.
140. Zhang L, Stoffel A, Behrisch M, Mittelstadt S, Schreck T, Pompl R, Weber S, Last H, Keim D. Visual analytics for the big data era—a comparative review of state-of-the-art commercial systems. In: Proceedings of the IEEE Conference on Visual Analytics Science and Technology, 2012. pp 173–182.
141. Harati A, Lopez S, Obeid I, Picone J, Jacobson M, Tobochnik S. The TUH EEG CORPUS: A big data resource for automated eeg interpretation. In: Proceeding of the IEEE Signal Processing in Medicine and Biology Symposium, 2014. pp 1–5.
142. Thusoo A, Sarma JS, Jain N, Shao Z, Chakka P, Anthony S, Liu H, Wyckoff P, Murthy R. Hive: a warehousing solution over a map-reduce framework. Proc VLDB Endowment. 2009;2(2):1626–9.
143. Beckmann M, Ebecken NFF, de Lima BSLP,Costa MA. A user interface for big data with rapidminer. RapidMiner World, Boston, MA, Tech. Rep., 2014. [Online]. Available: http://www.slideshare.net/RapidMiner/a-user-interface-for-big-data-with-rapidminer-marcelo-beckmann.
144. Januzaj E, Kriegel HP, Pfeifle M. DBDC: Density based distributed clustering. In: Proceedings of the Advances in Database Technology, 2004; vol. 2992, 2004, pp 88–105.
145. Zhao W, Ma H, He Q. Parallel k-means clustering based on mapreduce. Proceedings Cloud Comp. 2009;5931:674–9.
146. Nolan RL. Managing the crises in data processing. Harvard Bus Rev. 1979;57(1):115–26.
147. Tsai CW, Huang WC, Chiang MC. Recent development of metaheuristics for clustering. In: Proceedings of the Mobile, Ubiquitous, and Intelligent Computing, 2014; vol. 274, pp. 629–636.