Big Data Analysis Using Machine Learning Approach to Compute Data

Anubha Parashar[a,1], Apoorva Parashar[b] Somya Goyal[c]
[a]*Manipal University Jaipur, India,* [b]*Maharshi Dayanand University, Rohtak, India.*
[c]*Vaish College of Engineering, Rohtak, India,*

Abstract. When machine learning algorithms are applied to huge amount of data, we found difficulties to process such huge data. Now new approaches are being adopted because existing machine learning libraries doesn't have enough resources to process large datasets. So new libraries (CUDA, MapReduce, and Dryad) are adding up for concepts like parallel computing. Here we will take account of GraphLab, Apache MahoutTM, and Jubatus to get the exposure of famous academics and industrial results. Looking at the traditional machine learning techniques, tasks like to handle the data which is distributed identically or in batch mode becomes impossible and there is requirement to develop new algorithms to overcome with the existing difficulties faced by these traditional ML algorithms. The objective of this chapter is to provide overall view of developed algorithms and paradigm shifts of current big data analysis using machine learning approach to compute data. Here we will explore that the machine learning field has great impact on cloud computing paradigm. In first step we deploy various tool to the cloud like libraries and statistics tools. In second step we embed plugins with current tools in order to make Hadoop cluster on the cloud so that working programs can run on it. In third step libraries of machine learning algorithms are deployed and used for data intensive computing.

Keywords. Big data, Machine learning, 10Vs, Data Analytics, Cloud Computing

1. Introduction

With the fast growing technology data is more digital rather than manuscript. Digital data is very fast and easier to process as we share the information on internet. According to a study availability of digital media is much more than five Exabyte's. The problem of analyzing larger data is not from today but this problem exists from much before. In 1950s specifications of the hardware was not enhanced as compared to now so on the less efficient hardware software works slow. Though the technology is fast growing and the computers now-a-days are much faster still the problem persist as the data is growing larger and larger and to analyze such big data we need efficient algorithms.

There are many ways to analyze big data, but there are very less number of solutions that are efficient [1], like condensation of data, data sampling, density based approach, D&Q (divide & conquer). Emerging technologies such as machine learning includes many solutions and efficient algorithms using statistics and artificial intelligence [2]. Though machine learning is best way to perform data analytics in data science but

[1] Anubha Parashar, Computer Science and Engineering, School of Computing and Information Technology, Manipal University Jaipur, India; E-mail: anubhaparashar1025@gmail.com.

apparently machine learning also consumes time so lot of changes have been made in order to enhance the execution-time of ML algorithm shown in figure 1.

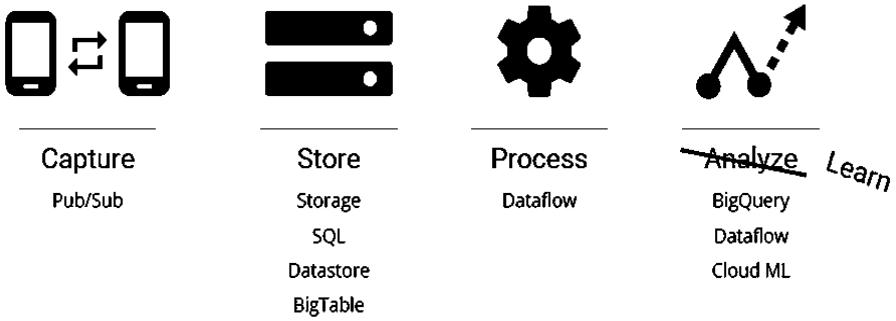

Figure 1. Big Data Lifecycle.

When we shift to cloud computing techniques then execution time will drastically decreases and we get enhanced performance. Mostly used tools of statistics in this field are Python, R, Octave and compatible with cloud computing as well. We have two options to use these statistics tools with cloud [3]. First is, in order to integrate statistics tools we make a cluster in cloud and then with statistic tools we bootstrap it. Second is, we make Hadoop cluster in cloud, then augmenting of plugins with statistic environment takes place and then with statistic tools we bootstrap it, finally jobs can be run on it.

Certain environments (Octave, R, Mapple) provides low level architecture for data analytics and cannot be applied to larger datasets but if cloud based technology is applied to these environments then same features can be used for huge datasets [4]. Now to enhance all this machine learning technique can be applied and it provides retrieval of data from large dataset that can be used for further analysis, as machine learning provides many knowledge and learning models, shown in figure 2.

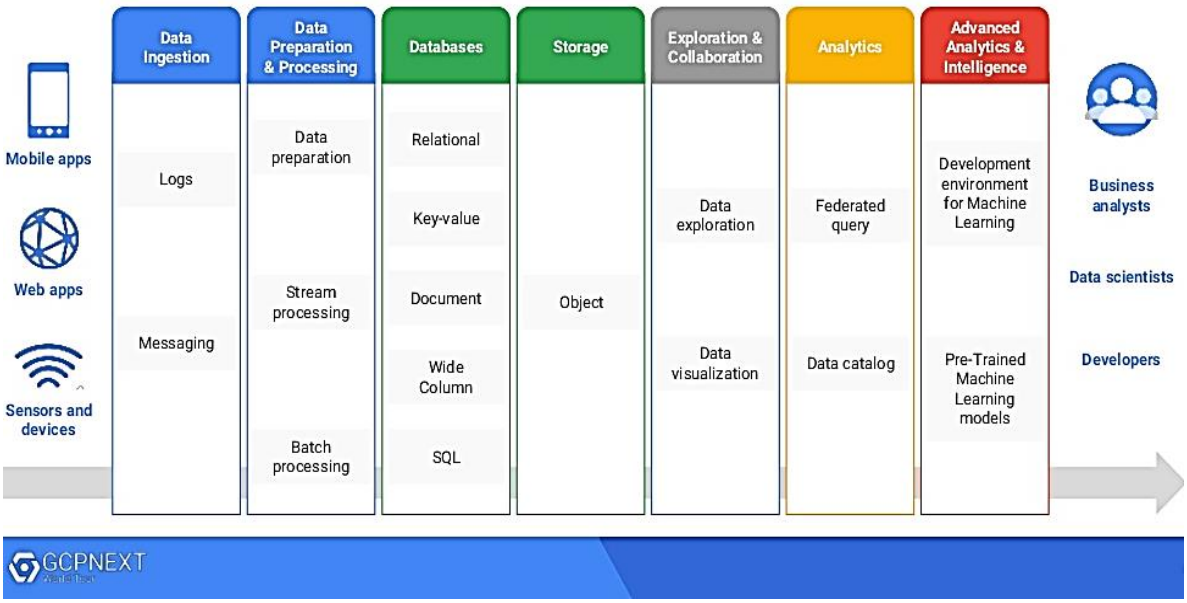

Figure 2. Data Analytics process using machine learning and cloud computing in data science.

1.1. Definition of Big Data

Big data describes to extremely huge quantity of data that need to be computationally analyzed which is generated by sensors, results of some scientific experiments, business records etc., shown in figure 3, that may organized (**structured data** which is organized in data base), unorganized (**semi-structured data** which is not organized in data base) or generic label (**unstructured data** which doesn't exist in data base or do not have data structure associated with it). Now this extreme huge data expects highly sophisticated and advanced technology in storing the data and then this data has capacity to be useful for mining the data, shown in figure 4 [5-9].

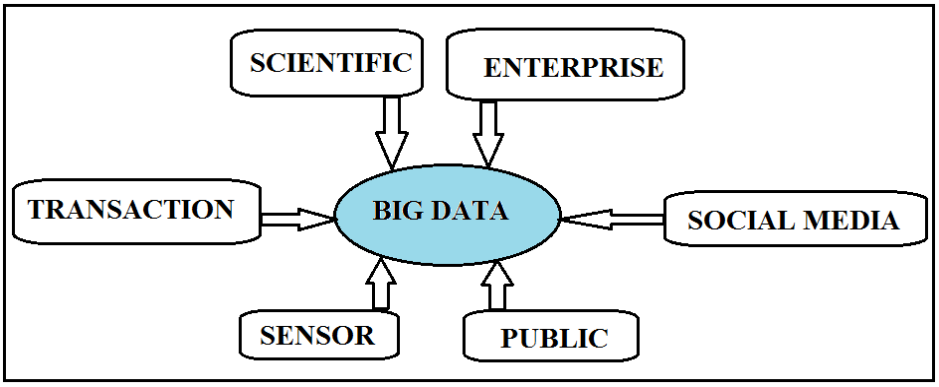

Figure 3. Source of generation of Big Data.

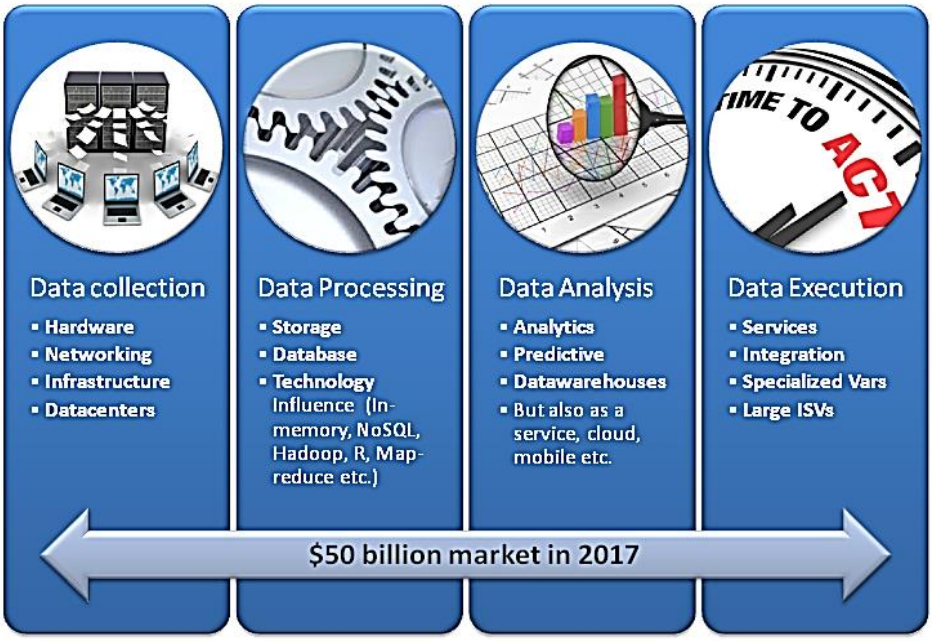

Figure 4. Landscape of Big Data.

1.2. Ecosystem of Big Data

Big data is diverse concept and it covers many areas both technical and non-technical. It is dynamic and any changes from user can be easily adaptable [10-14]. It is very efficient for emerging technologies like Hadoop, in memory computing, NoSQL and every day challenges are faced by these technologies, shown in figure 5.

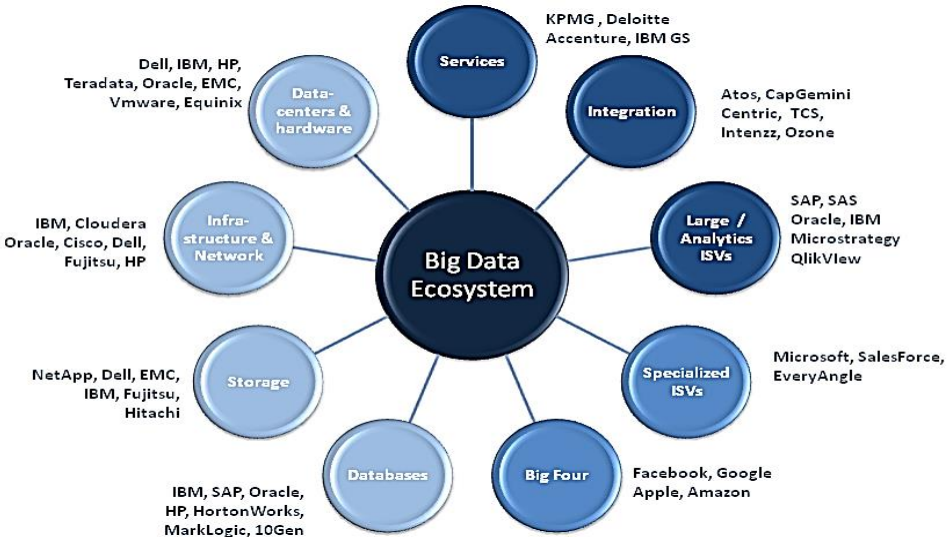

Figure 5. Landscape of Big Data.

1.3. 10 V's of big data

There are few characteristics of big data and those are Volume, Variety, Velocity, Veracity, Validity, Value, Variability, Venue, Vocabulary, and Vagueness shown in figure 6. These are known as 10 Vs of big data, according to Dough Laney big data just not focus on the mass/size of the data but also on these 10 factors [15-18], shown in figure 7.

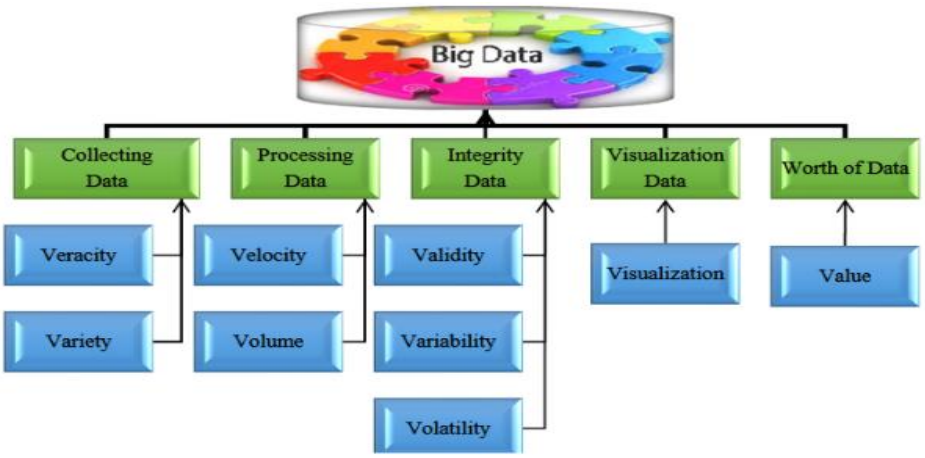

Figure 6. Categorization of Big Data.

Figure 7. 10 Vs of big data.

1.3.1. Volume

Volume represents huge amount of data. This data can be produced by machines, generated by networks or from interaction of humans on social media [20]. Day by day volume of data is increasing and according to IBM, every day 2.5 Exabyte data is getting generated [19]. Up to 2020 data volume will increase up to 40 zettabytes (40,000 Exabyte), shown in figure 8.

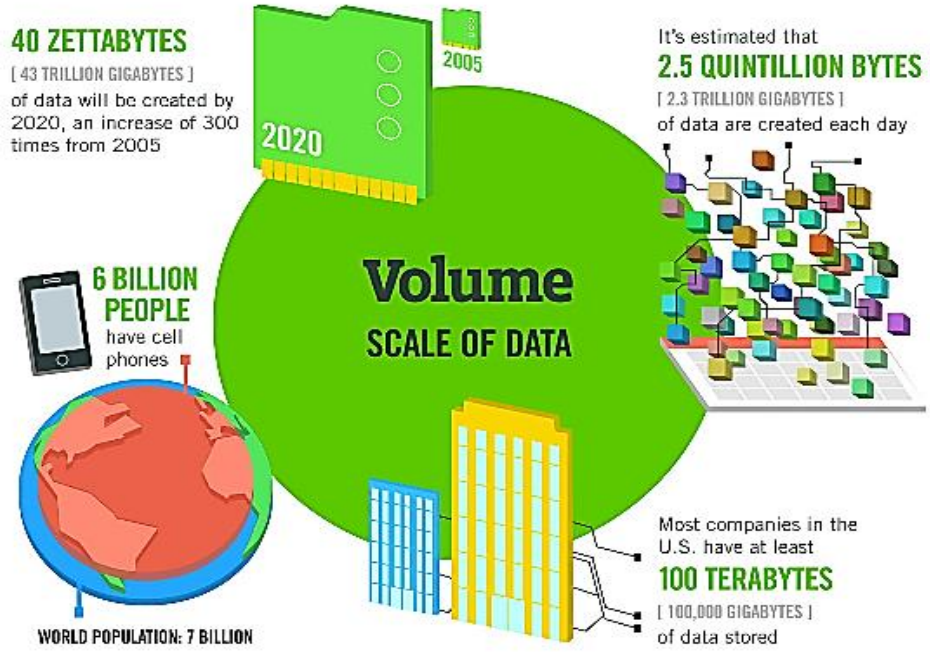

Figure 8. Volume of big data.

1.3.2. Variety

Variety means that there can data coming from many different sources and its type could be both unstructured and structured data. Storage space that we use are the databases and spreadsheets. But the data is coming from different sources like audio, video, photographs, emails, sensors, monitoring devices, PDF, social media, transactional data etc. shown in figure 9. As the data has so much variety and it is also unstructured therefore, it becomes difficult for storing, retrieving useful information (mining) and to analyze it [21-27]. As there are many features in the data therefore while using machine learning technique there are problems like combinatorial explosion, curse of dimensionality, there are various data formats and numerous data types [28].

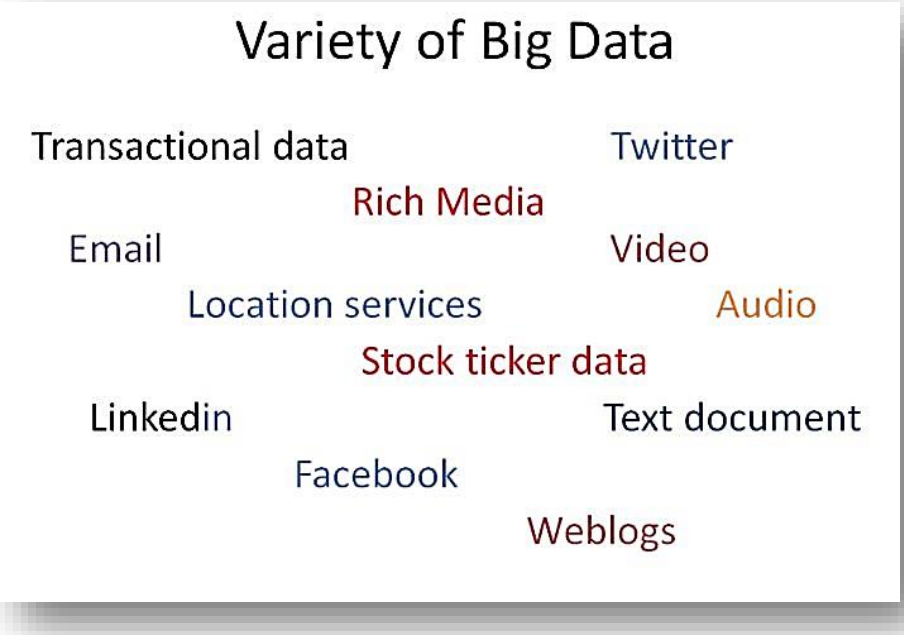

Figure 9. Variety of big data.

1.3.3. Velocity

Velocity refers to the speed (pace) of data at which it flows in various forms from different sources like machines, networks, business processes, mobile devices, social media, etc [29]. The real time flow of data is continuous and enormous and this data can be used for research purposes or for industries like for business purposes to come up to a conclusion in decision making [30].

In spite of such a great speed or the high velocity of the data coming from real time there has to be storing device/technology [31-39], it has to be analyzed/visualized, shown in figure 10.

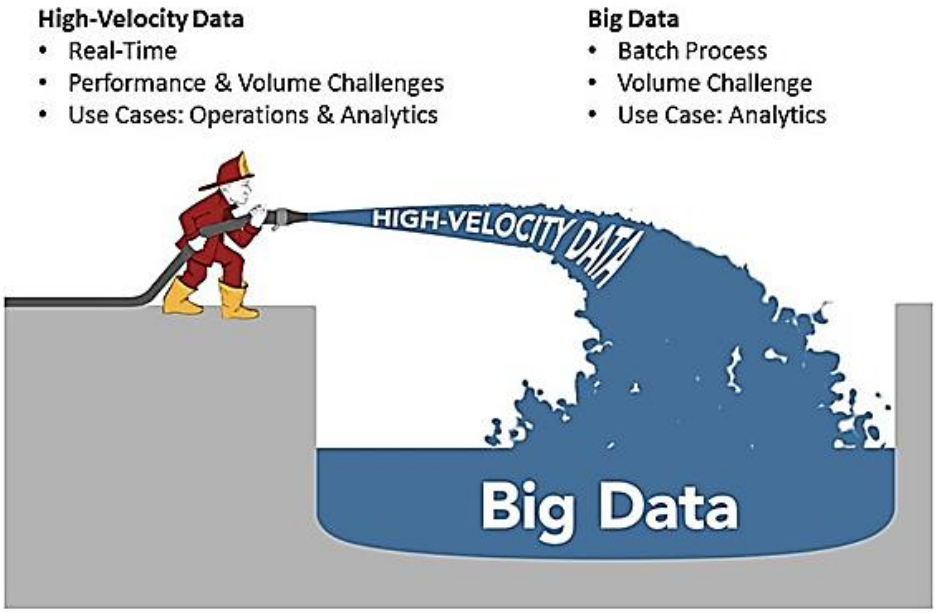

Figure 10. Comparing high velocity and big data.

1.3.4. Veracity

If we look into the past, structured data was created fitting exactly in rows and columns. But today 90% of data is unstructured, generated from many sources in all forms and shapes i.e. tweets to geo spatial data, that has to be analyzed to get information about the content and sentiments can be derived from it [40], shown in figure 11

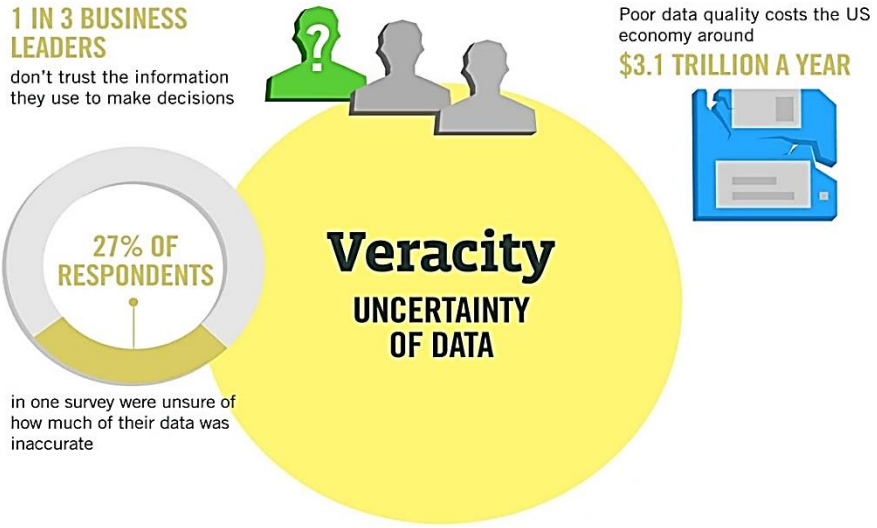

Figure 11. Comparing high velocity and big data.

1.3.5. Validity

When we talk about validity of data, we mean to check that weather the data is accurate and correct for further use [41]. The complete experimental concept is comprehended by validity. With the help of validity it becomes possible to check which of the results procured is apt for the method of scientific research shown in figure 12.

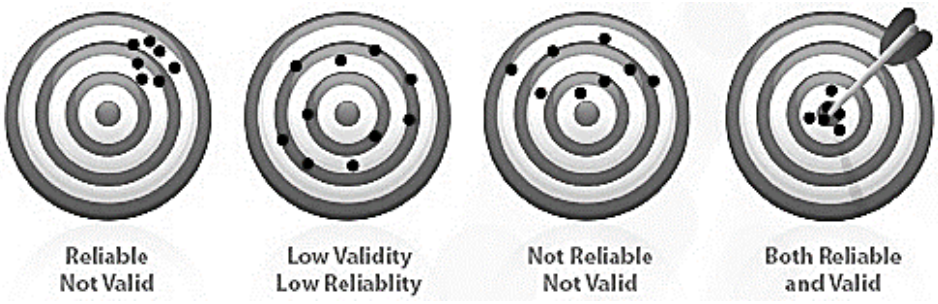

Figure 12. Validity of big data.

1.3.6. Value

When we talk about the value then the data we retrieve is not valuable but the analysis that is done on that data can be valuable if we are able to get useful information out of it. There can be different ways to get information out of data and making the system learn is the best way so we can use machine learning in this case [42]. Various tools and steps are shown in figure 13 in order to extract the valuable features.

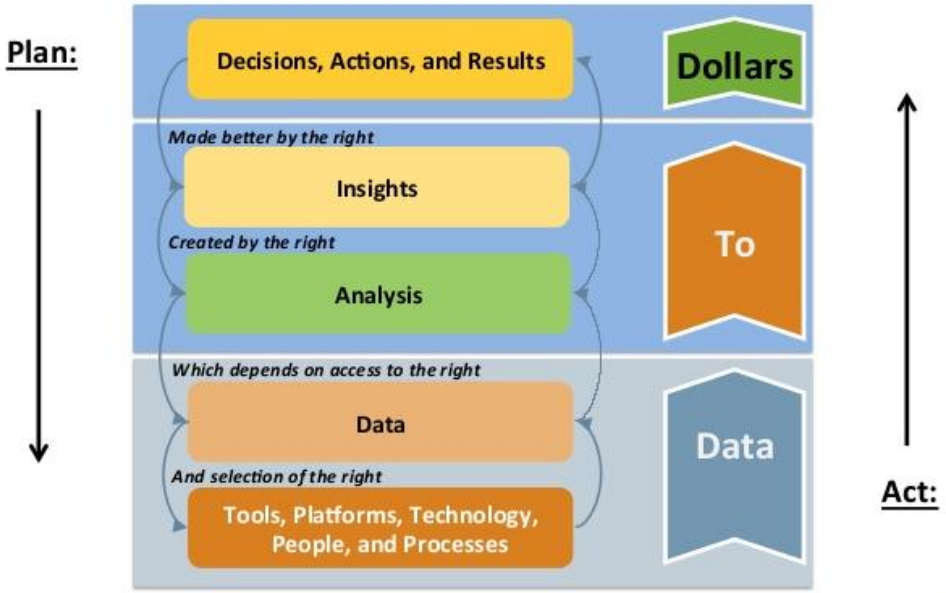

Figure 13. Stack is also a value chain in big data.

1.3.7. Variability

Muddling up of Variety and variability is very common. Let's understand the difference between these two with the help of an example. When we go to CCD (cafe coffee day) we find a large number of items in their menu. There are many types of shakes, ice-creams, mocktails etc. This is known as variety. Let's say that every time you go to CCD you drink blue lagoon and you feel that every time it tastes slightly different from the last time you had it. This is known as variability. In short variability means inconsistency of data, this can be seen in the figure 14, the varying of sales data and getting results on variability that how customers have varied on sale [43].

Figure 14. Variability in big data.

1.3.8. Venue

The data we gather here comes from various platforms, and this makes the data heterogeneous in nature as it is from different sources (owners), in numerous formats [44]. Therefore it becomes difficult to gather such data in structured format. The challenge is to make such unstructured data into structured format in order to access the data fast and retrieve the information so that we get knowledge as output, shown in figure 15.

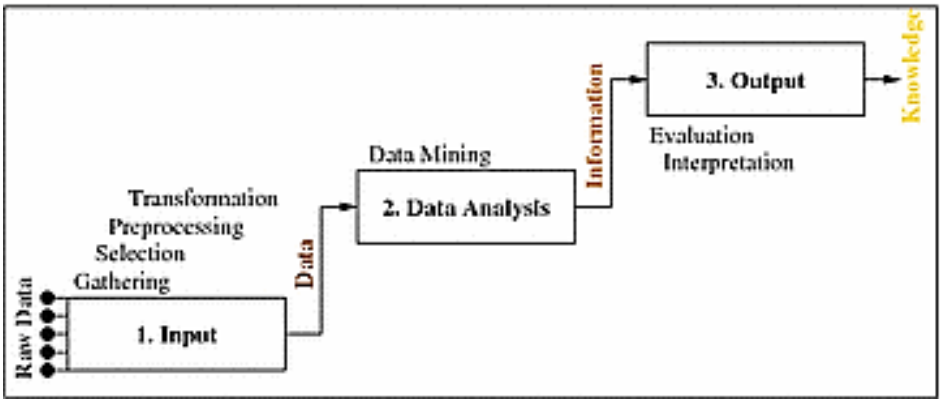

Figure 15. Venue in big data.

1.3.9. Vocabulary

Vocabulary defines all the data models that has been used to represent data, related semantics, taxonomies, ontologies, metadata based on content to define structure of data, syntax of data, content of data and its origin shown in figure 16.

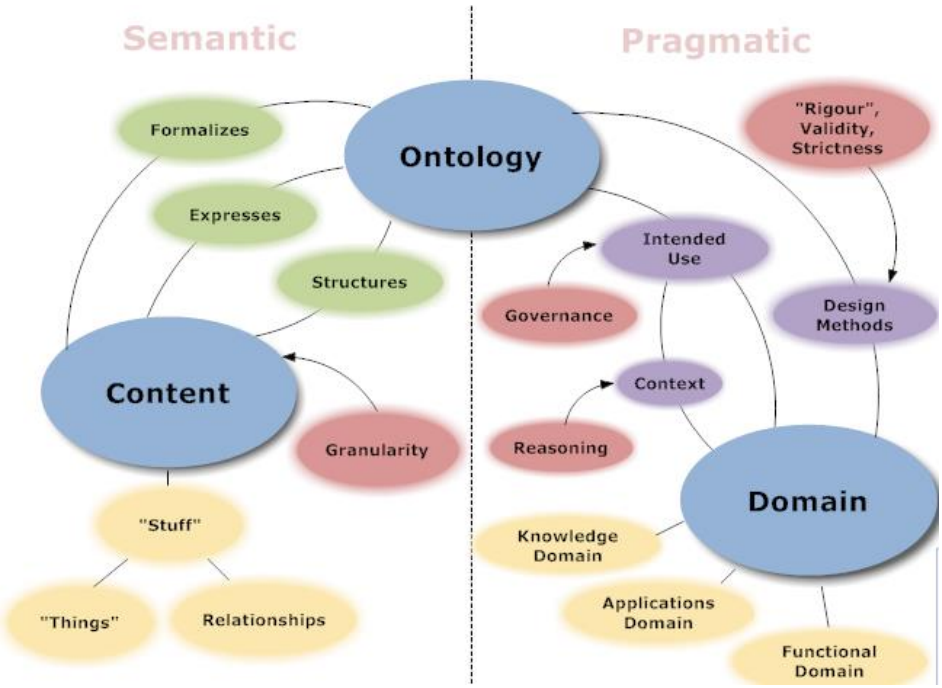

Figure 16. Venue in big data.

1.3.10. Vagueness

When we define the entire concept of big data, i.e. the gathering of data then mining of data, after which we apply an algorithm to make the system learn and train the network through artificial intelligence is processing about big data. But the concept is still vague as up to now there is still confusion weather the big data is Hadoop or is it a new technology or the existing one. What is new in the technology and the tools related to it will work or not. This is vagueness and shown in figure 17.

Figure 17. Vagueness in big data.

2. Methodology

Various steps of the workflow of big data analytics is shown in figure 18. Big Data is significantly large data set that may be examined computationally to show patterns and trends. Models are built by using data from different sources like data warehouses, databases. For building models data goes through the process of cleaning and filtering. Once the data is refined it can be used to train data models. Lastly the model is applied on data.

Data Mining is the computing process of discovering patterns in large data set involving methods. It is used to solve problems by analyzing data. Data mining is used to turn raw data into meaningful information. For this the data goes through refinement process.

Fig. 18 depicts the common phases of a traditional analytics workflow for Big Data. Big Data is significantly large data set that may be examined computationally to show patterns and trends. Models are built by using data from different sources like data warehouses, databases. For building models data goes through the process of cleaning and filtering. Once the data is refined it can be used to train data models. Lastly the model is applied on data.

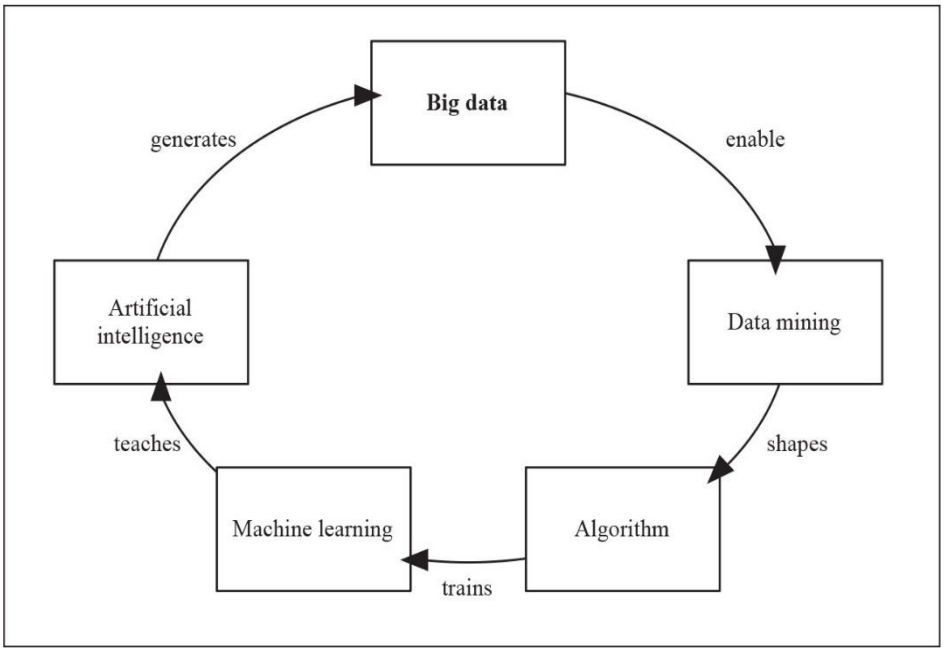

Figure 18. Vagueness in big data.

Data Mining is the computing process of discovering patterns in large data set involving methods. It is used to solve problems by analyzing data. Data mining is used to turn raw data into meaningful information. For this the data goes through refinement process.

Earlier historical data was used in identification of patterns and creation of management reports. But using this approach was not a good choice as current analytics are based on software systems built for general purpose. Hence, integration of data sources is done to make data useful for specific requirements of organizations. These solutions are generally complex and their operations can take hours to get executed as the data is very large. This is where cloud computing comes into play. With the help of cloud computing we can get faster access of our data. We can host the solutions on the cloud. But for hosting data on cloud we need to address problems like privacy, data quality, data management, tuning of data models and data currency. We mainly focus on key features in states of analytics solution. Most of the problems in big data are related to data management, integration and processing. Security is a key challenge for hosting analytics solutions on public Clouds as all the data is stored on the cloud.

In figure 18 we have done work on addressing challenges on cloud environments, and have also elaborated models to provide data on cloud. We have also given solutions for visualizing data and of interacting with a customer with the help of analytics solutions. Some of the challenges related to business models, service level agreements and service structures are also discussed.

The paper is divided into following sections:

- Data
- Compute Infrastructure
- Storage Infrastructure

- Analytics
- Visualization
- Security and Privacy
- Conclusion

3. Data

Data part is very important factor in big data, as when the huge data is gathered then there is requirement for categorizing the data and its domains, in order to get the requirements and understand architectural choices which is required for certain data types. The data which we gather is not equal. Looking at the architecture of data we can find out the types of infrastructure that is required to store data, process the data and then to do data analytics on it.

3.1. Requirements

3.1.1. Real time

Hardware and software are concerned with real time constraints in real time computing (RTC). Deadlines are present in real time computation. Deadlines here refer to response within specified time constraints. The accuracy of these systems depend on their time related and functional features.

A real time system does all the work like controlling the data by receiving, processing and returning the results in time. Operations in real time systems need to be responsive and should not get interrupted by any other process. That is, there should not be any other process running in the background except the one we are currently working on. Real time computing is classified in following three types-

- Soft – In this type of real-type system, the quality of service gets degraded as after hitting the deadline the functionality of the result gets reduced.
- Firm – In this type of real-type system the quality of the system gets degraded with irregular deadline misses. The functionality of the result becomes zero after the deadline.
- Hard – In this type of real-type system the system crashes i.e. becomes unresponsive in case it misses the deadline.

3.1.2. Near Real Time

It means that the system is exclusively Real-time but such a system does not assure of completing the work before specific deadlines. Near Real Time is also known as soft real time (as opposed to hard real time).

3.1.3. Batch Processing

In case of batch processing latency can be tolerated in larger amount and the reason behind this is in few seconds the results need not to be produced.

3.2. Structure

In order to map different domains of big data is

- structured data: which is organized in database
- Semi-structured data: which is not organized in data base
- Unstructured data: which doesn't exist in data base or do not have data structure associated with it.

Difference between structured, semi structured and unstructured data is shown in figure 19.

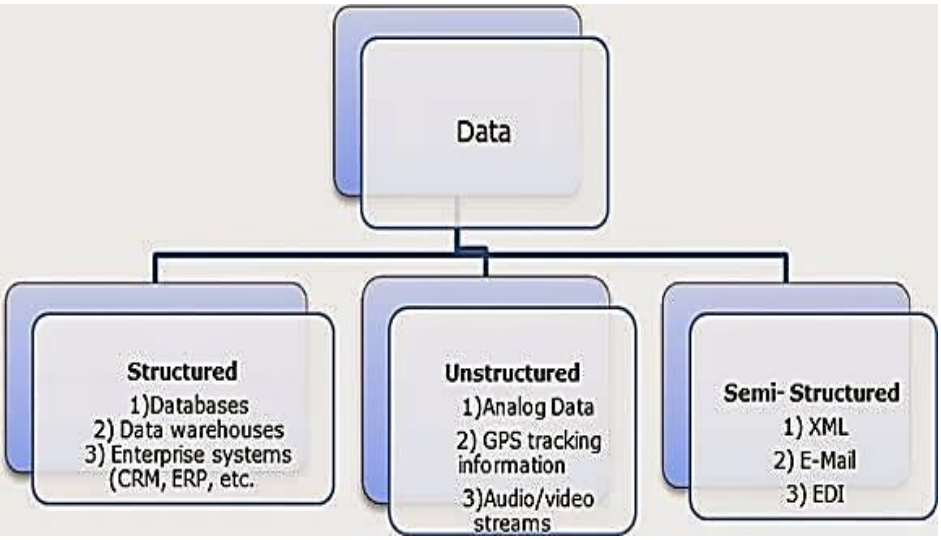

Figure 19. Structured data VS semi structured VS unstructured data.

The inputs of big data are known as data domains. The domains and sub domains of the big data are shown below in the figure 20.

Figure 20. Domains of Data.

Big data is mapped vertically to the time and organization axis. The vertical mapping of data is shown in the figure 21 below.

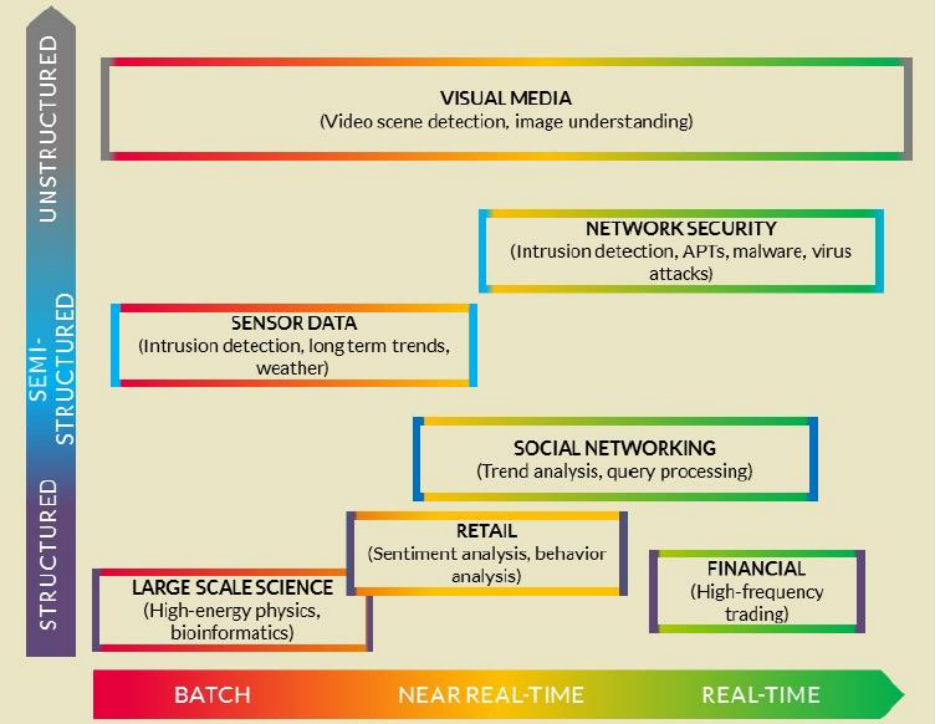

Figure 21. Big data mapping.

4. Compute Infrastructure

Several frameworks are available which can be used in various ways for computing but Hadoop ecosystem is one of the most popular choice for managing a very large dataset.

In Figure 22 we have depicted various approach of processing design. There is a level of abstraction on first level of computation of big data. There may be real time/ near real time on streaming of data or the processing might have been done in batch mode. We are depicting two frameworks here- for real time processing we are using spark and we are using Hadoop for batch processing.

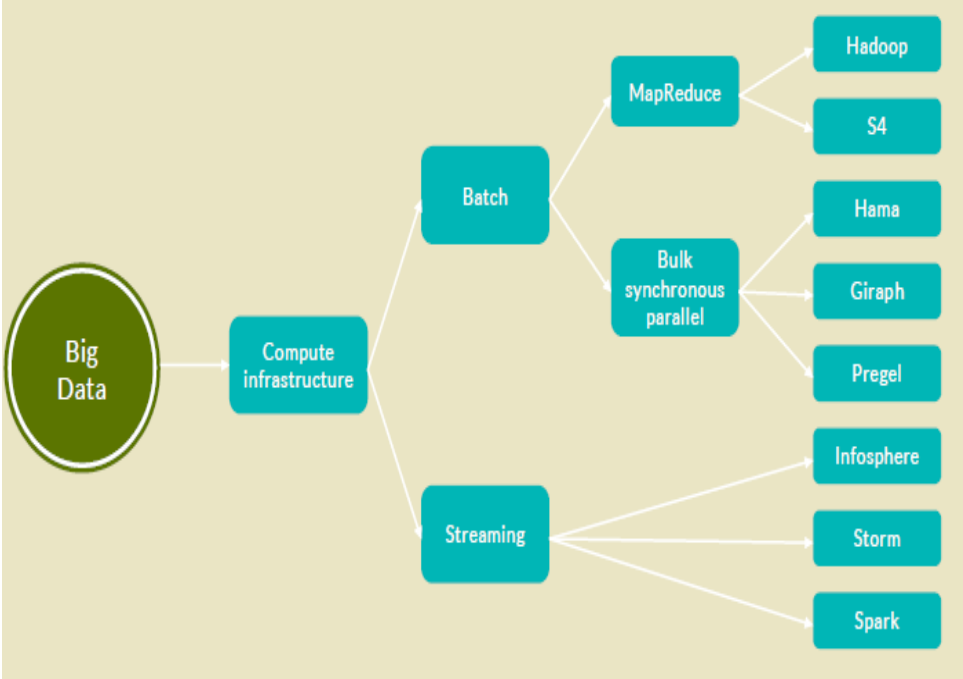

Figure 22. Computational architecture

MapReduce is a programming model and a processing technique. It is an algorithm and is based on java. It is used for generating large dataset and for processing data within a dataset. It has two parts- Map and reduce.

Map takes a dataset and converts it into different dataset. The dataset breaks into tuples. The reduce part takes these tuples and further breaks them into smaller tuples. When we take input, it passes through Hadoop and gets stored in map.

4.1. Batch Processing

In case of batch processing latency can be tolerated in larger amount and the reason behind this is in few seconds the results need not to be produced. Therefore batch oriented methodology is fairly adoptable. If we use Hadoop then batch processing is used efficiently.

4.2. Hadoop 1.0

Hadoop is popular worldwide upcoming technology which is open source and supports distributed programming. Entire Hadoop is based on computing technique known as MAP REDUCE.

In map reduce technique input is first mapped in order to split the working nodes which are working on the inputs that are independent subsets and then we reduce the

solutions from the mapped sub problems. Then it is combined in order to get output of entire computing, shown in figure 23.

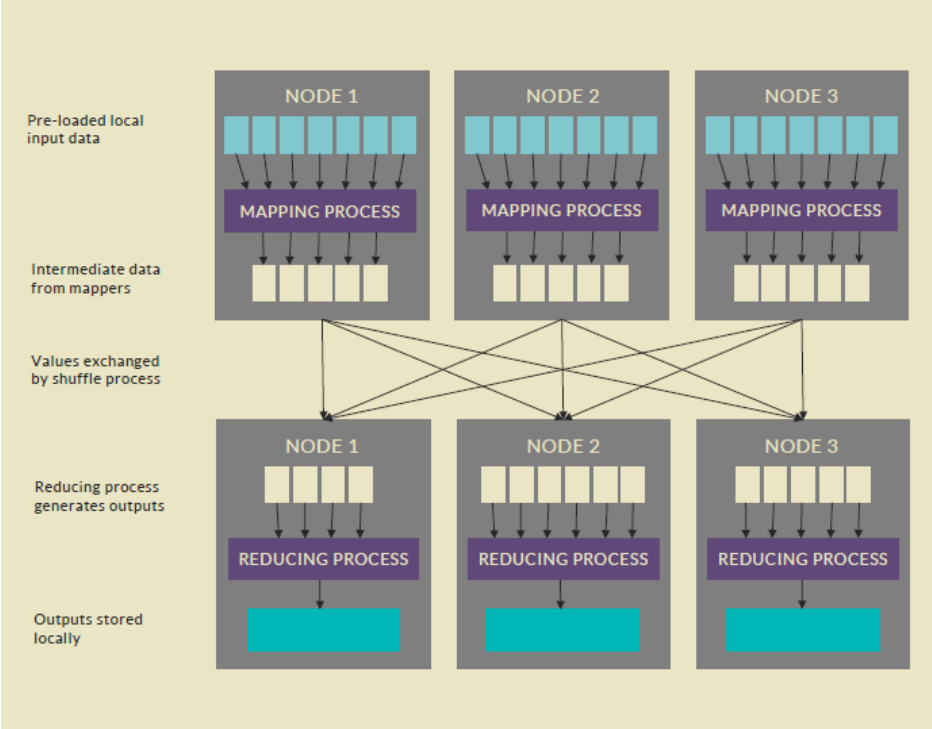

Figure 23. Procedure of Map-Reduce

Though Hadoop is found to be good at batch processing but for data streams it is not suitable which are not terminating, reason behind is in Hadoop, job reckon the data is in files of different nodes and therefore MapReduce will start on fixed inputs to give fixed outputs. Figure 24 depicts the various features and elements of Hadoop.

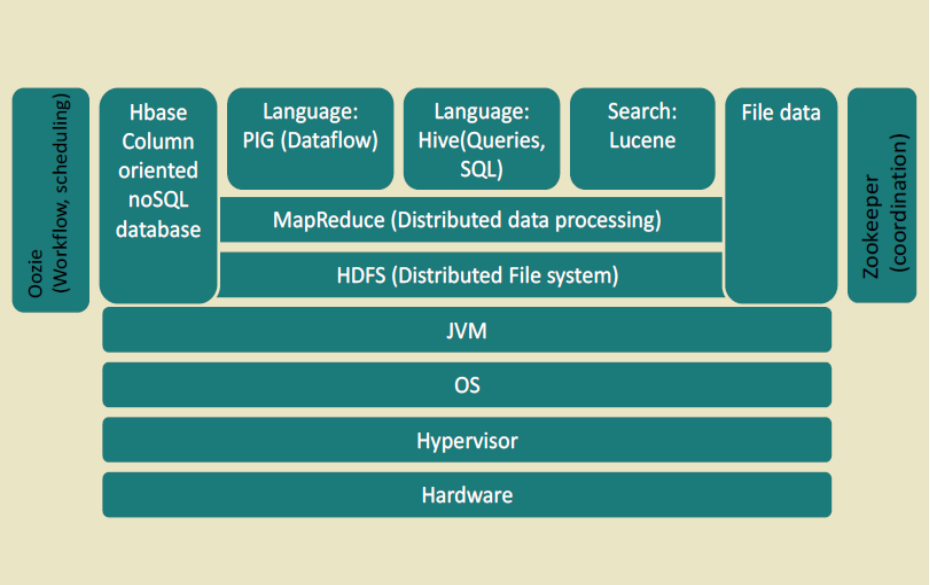

Figure 24. Hadoop 1.0 elements

Limitations of Hadoop 1.0

- Not suitable for data streams
- Not suitable for iterative algorithms as it includes machine learning algorithms and increase complexity in data analytics.
- Not suitable for shared global state algorithms as Map Reduce architecture is based on independent maps task which runs parallel and does not require access the state which is shared which will serve as bottleneck for performance due to delays in network, lock, semaphores etc.

4.3. Hadoop 2.0

As we can see lot of limitation in Hadoop 1.0, we are moving to Hadoop 2.0. This architecture has overcome the problems like managing the resources, Map Reduce programming. These things are achieved by introducing a new layer known as YARN which is resource managing layer which keeps track of resources at lower level.
Figure 25 depicts the various features and elements of Hadoop 2.0.

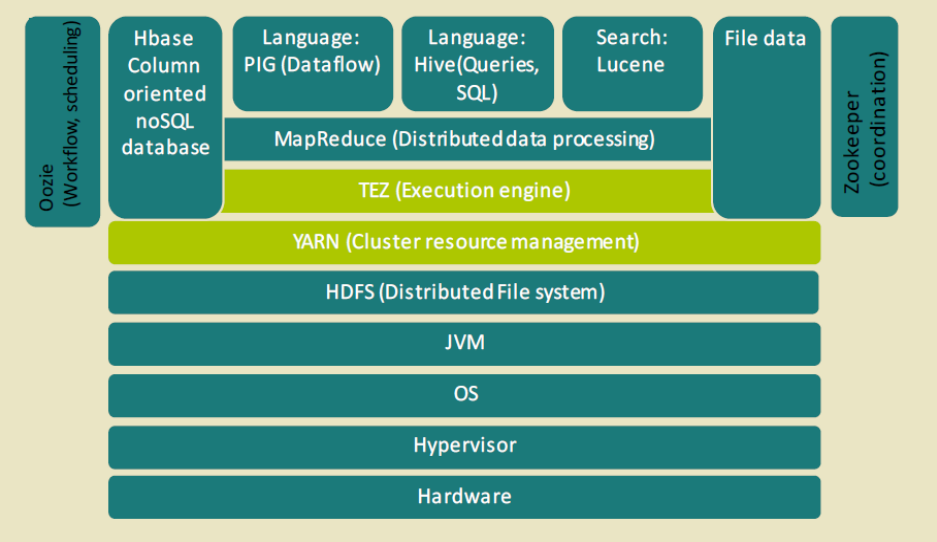

Figure 25. Hadoop 2.0 elements

4.4. Berkeley Spark

Berkeley Spark is again an open source database that is used to speed up data analytics process at development time as well as run time. The core idea behind Spark is resilient distributed dataset, i.e. when objects are collected then they are extended over a cluster which is stored in RAM/disk. Spark has fast implementation of programming language because its API is in Java, Scala, and Python. Spark can be used with the Hadoop 2.0, shown in figure 25.

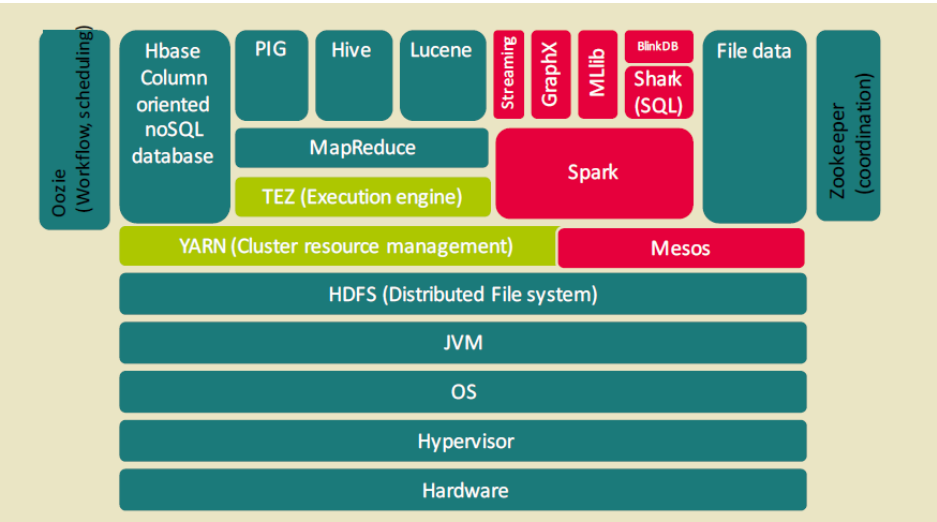

Figure 25. Berkeley Spark with Hadoop 2.0

5. Storage Infrastructure

The storage requirement of big data is that of type 'no limits'. For storing such data legacy infrastructure storages are of no use. Scale out storage can be considered for storing such vast data as maintaining performance of the system and providing rapid answers when it is queried is the foremost goal of such systems. It means we need to add more RAM or increase the capacity of HDD. In figure 26 we have shown the classification of different databases used to store big data [43].

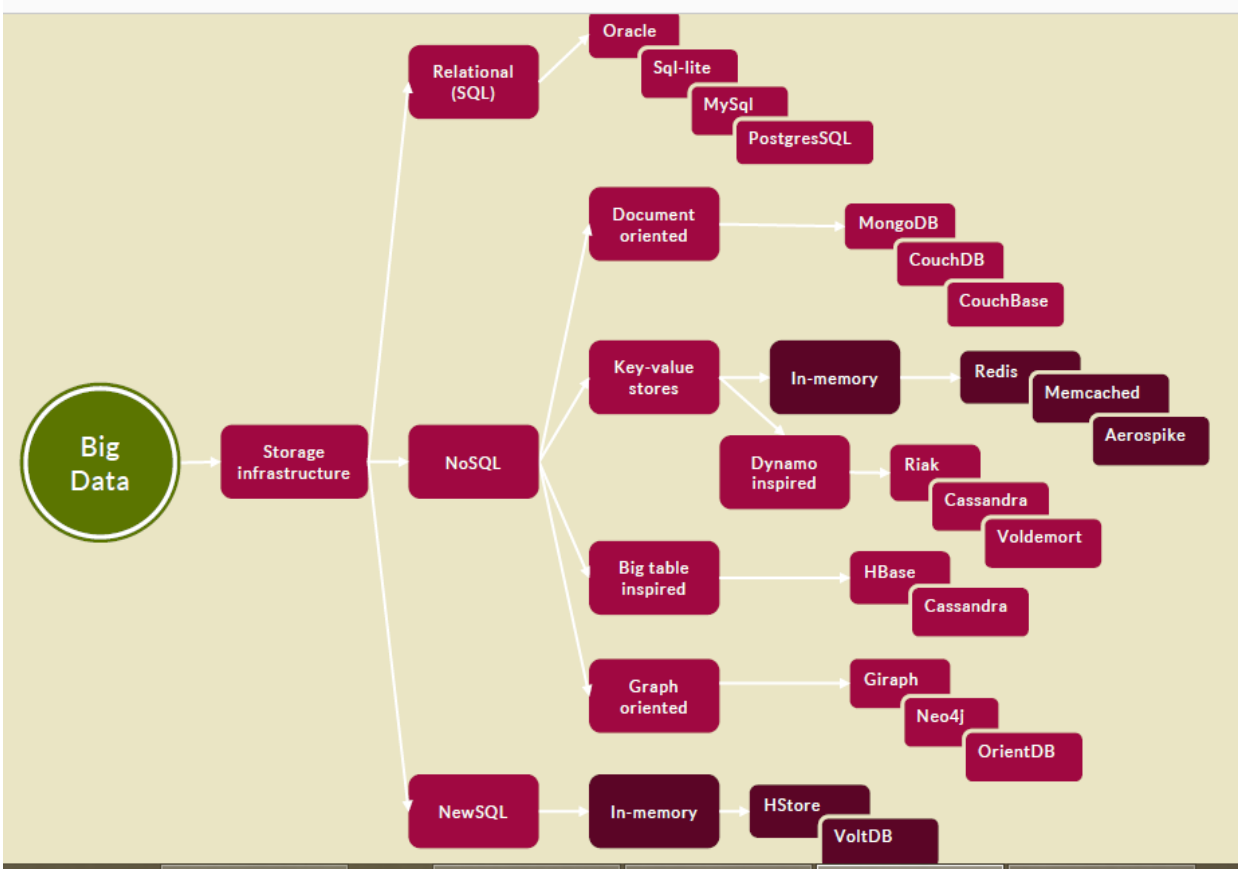

Figure 26. Infrastructure storage

In order to figure out information stored in database we use 10 V's, i.e. Volume, Variety, Velocity, Veracity, Validity, Value, Variability, Venue, Vocabulary, and Vagueness. In order for scaling process to be more effective we scale it across multiple servers in horizontal manner instead of upgrading a single server by scaling it vertically [44].

ACID properties define the level which helps in deciding the databases. These properties ensures that all the transactions made in the database are correct or not. ACID properties consist of following four properties-

- Atomicity- It means that if a transaction is taking place then it must get completed. If there is any problem in the transaction process I midway then the transaction completely stops i.e. the transaction fails and no changes are made in the database.
- Consistency- It means that the transaction currently taking place is consistent i.e. if a change is being made in one place and a copy of that same data is stored in some other place then consistency ensures that same changes are being made in all the places at same time.
- Isolation- This state of transaction keeps check of the current transaction that is being held. It ensures that if a transaction is taking place then no other transaction can be executed relating to the one currently in execution. That is, all the transactions should be held in serial order.
- Durability- Durability in transaction means that if we have made some changes in the data base i.e. any transaction was held and just after that the system resulted in failure then the stage till the transaction was fine should be saved. In short the system must get committed to the last transaction held and must remain committed even after system failures.

In relational Databases the ACID properties are a traditional approach. ACID properties of transaction are not enough to fulfill the requirements (fast response time and speed) and for storing big data on cloud. Therefore, Eric Brewer came up with a design philosophy called BASE [45].
BASE stands for-Basically Available, Soft state, eventually consistent. Today most of the databases are constructed using BASE.

Common database models, by their strengths are given below, also shown in figure 27.

- Relational database model- This type of database stores data in rows and columns.
    - Example- Oracle, SQLite, PostgreSQL, MySQL, VoltDB
- Document- This type of database Stores data in documents.
    - Example- MongoDB, CouchDB, BigCouch, Cloudant
- Key-value- This type of database Stores an arbitrary value at a key.
    - Example- CouchBase, Redis, PostgreSQL HStore, LevelDB
- Big Table-Inspired- In this type of database, Data put into column-oriented stores inspired by Google's BigTable
    - Example- Hbase, Cassandra (inspired by both BigTable and Dynamo)
- Dynamo-Inspired- In this type of database, Distributed key/value stores inspired by Amazon's Dynamo
    - Example- Cassandra, Riak, BigCouch
- Graph- This type of database Uses graph structures with nodes, edges, and properties to represent and store data.
    - Example- Neo4j, OrientDB, Giraph, Titan
- NewSQL- This type of databases stores data Like relational, except these databases offer high performance and scalability while preserving traditional ACID notions.
    - Example- VoltDB, SQLfire

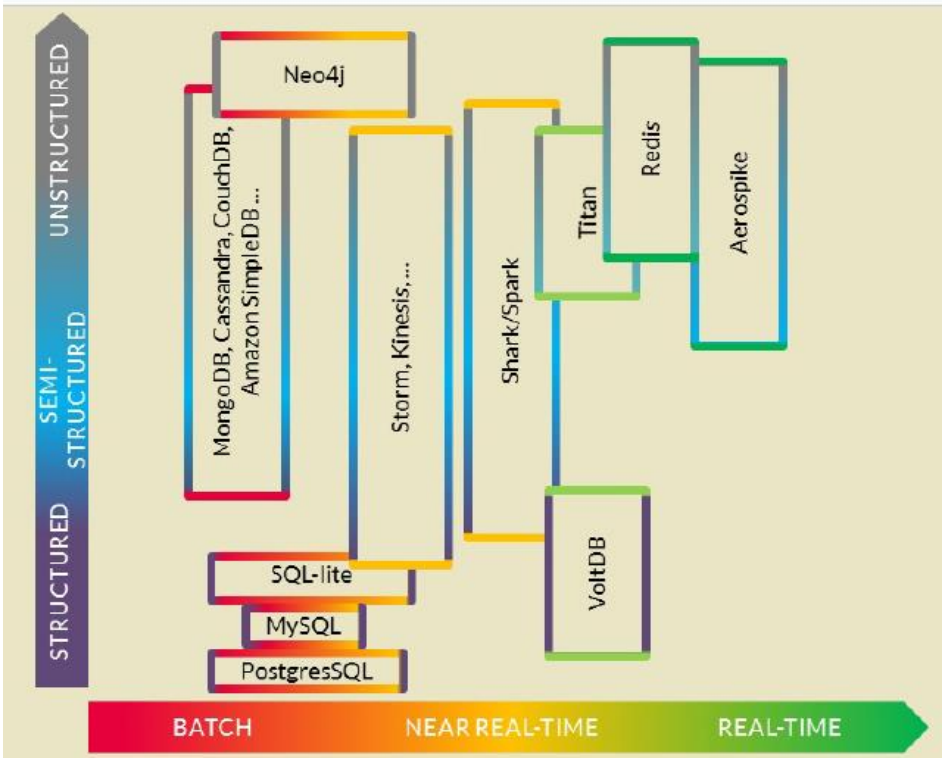

Figure 27. Map of different technologies providing storage

6. Analytics

6.1. ML Algorithms

Machine Learning is the skill that helps computers to make inferences from data and learn patterns on its own accord i.e. the machine need not be programmed explicitly [46]. These algorithms are classified along many different axes.

6.2. Different types of Machine Learning Algorithms

Tasks in Machine learning are classified into four broad categories. These categories depend on the nature of the learning or "feedback" or “signal” available to a learning system [47]. These categories are-

- **Supervised Learning-** The main aim of supervised learning is to learn a common guideline of mapping inputs to outputs.

- **Unsupervised learning**: In this type of learning techniques no labels are given to the learning algorithm, so the machine has to find structure of its input on its own.
- **Reinforcement learning**: A computer is required to perform certain goals when it interacts with a dynamic environment. As the program traverses its problem space, feedback is provided in terms of rewards and punishments.
- **Semi-Supervised Classification** – In order to resemble an appropriate learning algorithm, Semi-Supervised classification uses small groups of labeled data and combines this information with large data.

Different machine learning algorithms are shown in figure 28.

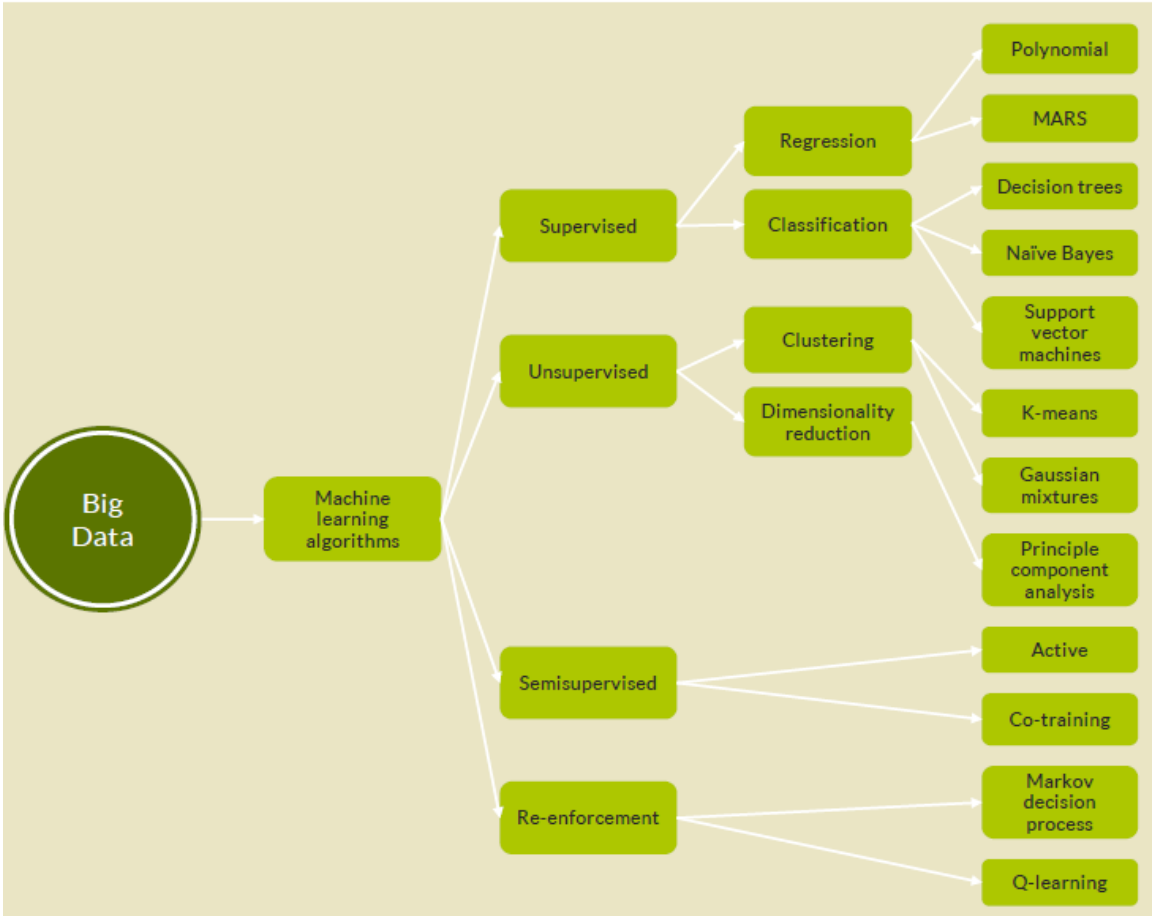

Figure 28. Algorithm on machine learning

6.3. Statistical Techniques

With the help of Statistical Techniques we can analyze big data very easily. Fig 29 provides various notions of machine learning and Statistical Techniques concept [49].

MACHINE LEARNING	STATISTICS
Network, Graphs	Model
Example/instance	Data point
Label	Response
Weights	Parameters
Feature	Covariate
Learning	Fitting/Estimation
Generalization	Test set performance
Supervised learning	Regression/Classification
Unsupervised learning	Density estimation, Clustering

Figure 29. Different statistical technique

Various machine learning algorithms have been mentioned [50] in figure 28, their flow chart and working is shown in figure 30.

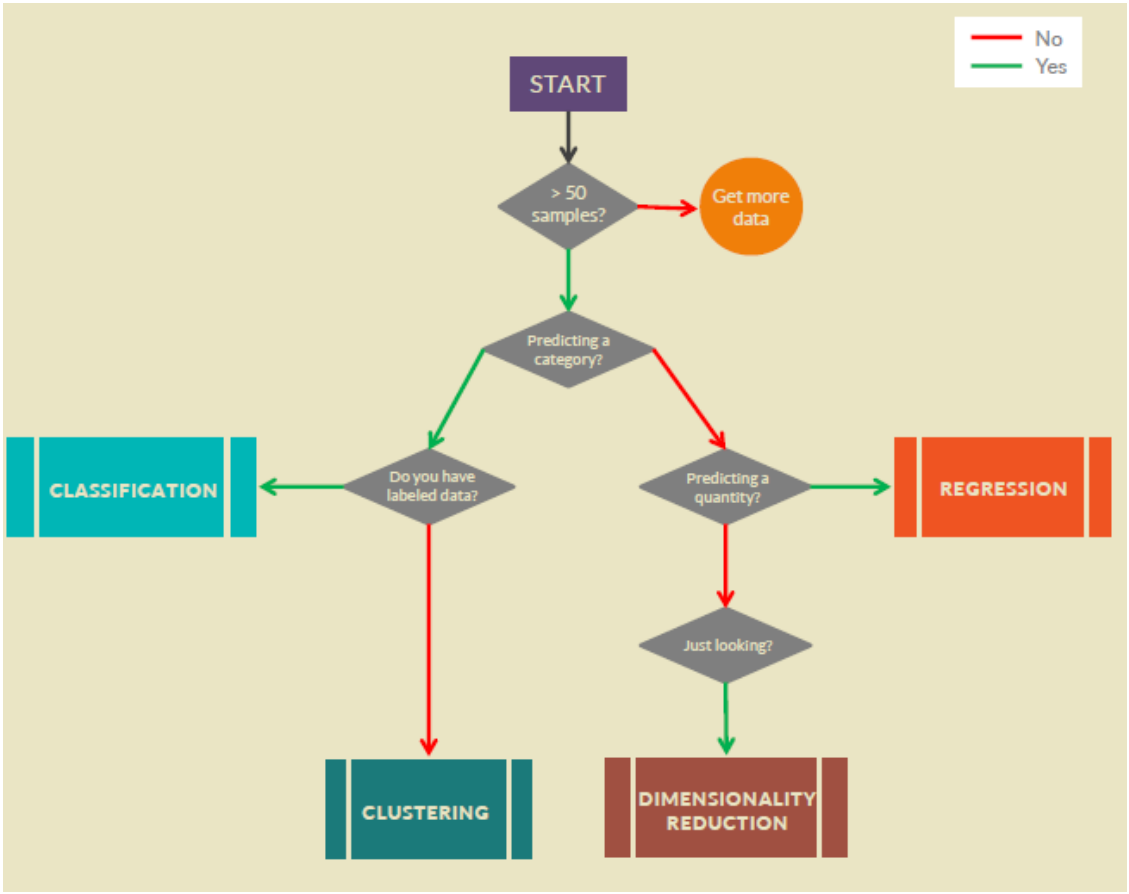

Figure 30. 4.4. Flow charts for machine learning

7. Visualization

The presentation of data in pictorial or graphical format is known as visualization, entire process is shown in figure 30.

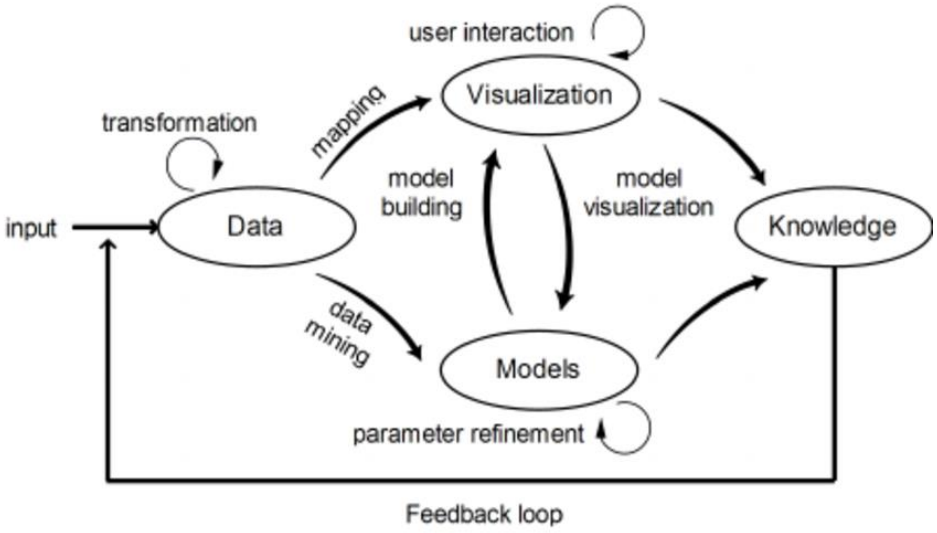

Figure 30. Visualization process

The techniques of visualizing big data can be classified in following three categories:

- Spatial Layout Visualization –To map a data object to a specific point on the coordinate space in a distinct way, we use spatial layout visualization. Some of the commonly used techniques for visualizing data using spatial layout visualization are- scatter plots, bar charts, line charts, etc.
- Abstract/Summary Visualization – Scaling of subsisting visualization techniques of big data become significant. Therefore, a new category of visualization techniques has been developed recently that processes and summarizes large-scale data before representing it in the process of visualization.
- Interactive/Real-Time Visualization – This technique is more robust as it allows the user to quickly judge the data. A fresh class of techniques are categorized under interactive visualization. These techniques interact with the user in real time. Such techniques makes it necessary for complex visualization mechanisms to take very less time for user to traverse the data [51-57].

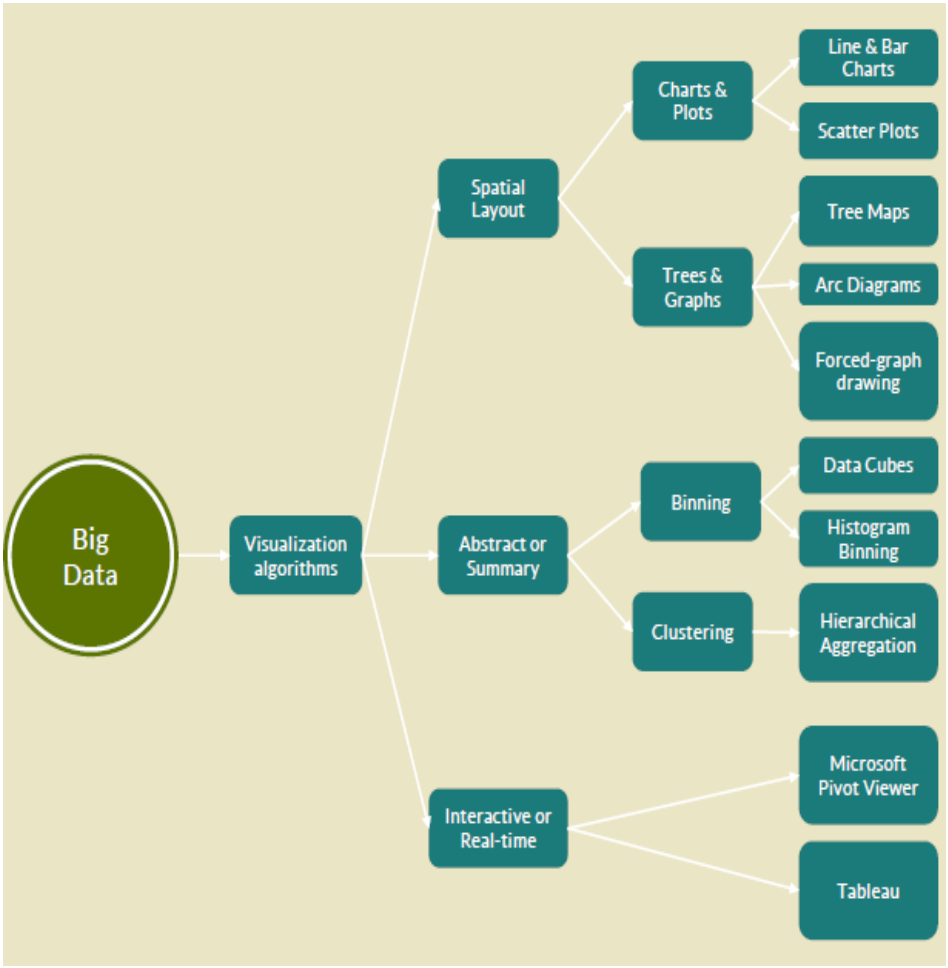

Figure 31. 4.4. Various visualization technique

8. Security and Privacy

Privacy and security is a topic of major concern when it comes to big data. Distributed computations and data stores are of main focus when it comes to security [58]. Therefore, methods like cryptography and granular access control are used for ensuring the security and privacy of important and sensitive data, various steps for classification of security & privacy can be seen in figure 32.

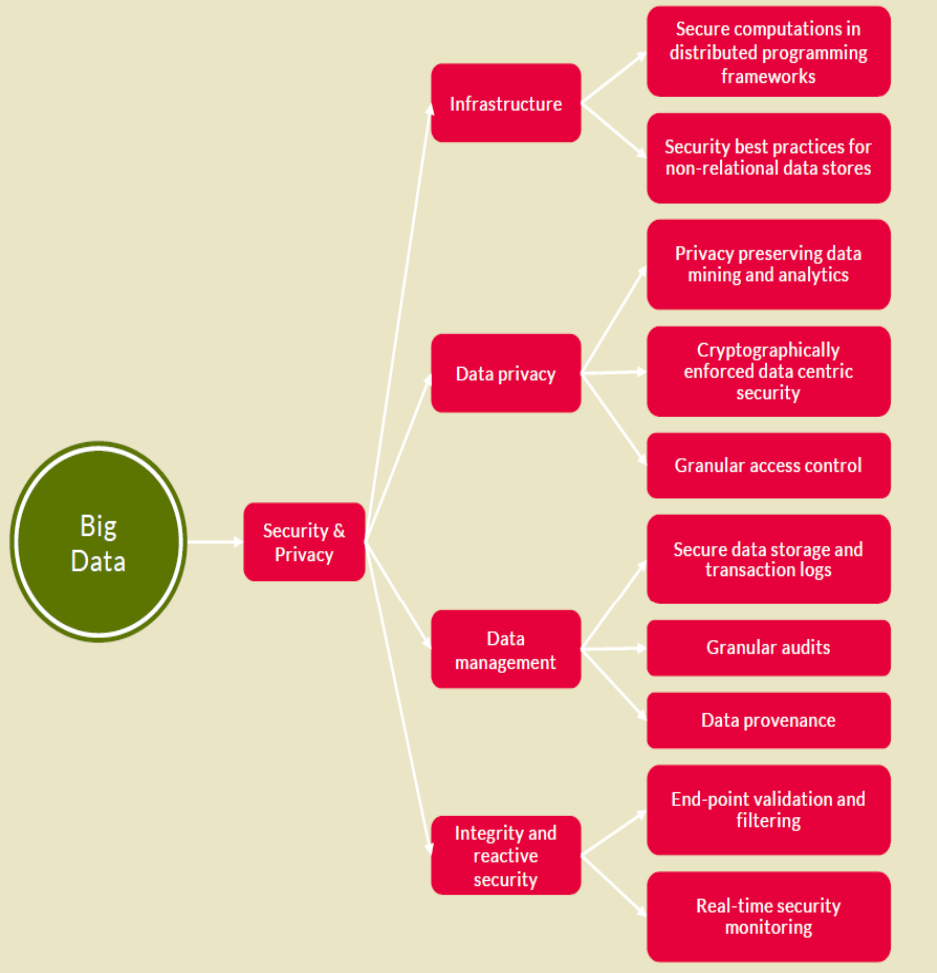

Figure 32. Various steps for classification of security & privacy

9. Conclusion

These days there is not only hype about big data but also now a days it has become the basic requirement. We have discussed in the chapter, how to gather data, what are different sources of data, how the computing infrastructure works, how the storage architecture works, then we analyzed the data using machine learning, then visualization, security, and privacy factors were discussed. Though the big data term is recently introduced but this technology is in use since many years. We have seen in the chapter that big data can only be analyzed using cloud but the system will not intelligent until we train the system using machine learning technique. Machine learning technique combine with cloud computing provides easier computation on big data.

References

[1] Xu R, Wunsch D. Clustering. Hoboken: Wiley-IEEE Press; 2009.
[2] R. Bekkerman, M. Bilenko and J. Langford (editors) { Scaling up Machine Learning, Cambridge University Press, 2012, summary at http://people.cs.umass.edu/~ronb/ scaling up machine learning.htm
[3] BIG DATA WORKING GROUP Big Data Taxonomy, September 2014.
[4] IBM Study, "StorageNewsletter » Every Day We Create 2.5 Quintillion Bytes of Data," 21-Oct-2011.
[5] http://www.storagenewsletter.com/rubriques/market-reportsresearch/ibm-cmo-study/.
[6] J. Dean and S. Ghemawat, "MapReduce: simplified data processing on large clusters," in Proceedings of the 6th conference on Symposium on Opearting Systems Design & Implementation - Volume 6, Berkeley, CA, USA, 2004, pp.10–10.
[7] L. Valiant, "A bridging model for parallel computation," CACM, vol. 33, no. 8, pp. 103–111, Aug. 1990.
[8] "Welcome to ApacheTM Hadoop®!" http://hadoop.apache.org/.
[9] S. Ghemawat, H. Gobioff, and S.-T. Leung, "The Google file system," in Proceedings of the nineteenth ACM symposium on Operating systems principles, New York, NY, USA, 2003, pp. 29–43.
[10] J. Lin and C. Dyer, Data-Intensive Text Processing with MapReduce. Morgan & Claypool, 2010.
[11] "Hadoop Tutorial - YDN." http://developer.yahoo.com/hadoop/tutorial/module4.html#dataflow.
[12] "Breaking Down 'Big Data' – Database Models – SoftLayer Blog." http://blog.softlayer.com/2012/breaking-downbig-data-database-models/.
[13] "CAP theorem," Wikipedia, the free encyclopedia. 16-Aug-2013.
[14] "CAP Twelve Years Later: How the 'Rules' Have Changed." http://www.infoq.com/articles/cap-twelve-years-laterhow-the-rules-have-changed.
[15] "Google Research Publication: BigTable." http://research.google.com/archive/bigtable.html.
[16] "Amazon's Dynamo - All Things Distributed."
[17] http://www.allthingsdistributed.com/2007/10/amazons_dynamo.html.
[18] "Graph database," Wikipedia, the free encyclopedia. 21-Aug-2013.
[19] "New SQL: An Alternative to NoSQL and Old SQL for New OLTP Apps." http://cacm.acm.org/blogs/blogcacm/109710-new-sql-an-alternative-to-nosql-and-old-sql-for-new-oltp-apps/fulltext.
[20] Max De Marzi, "Introduction to Graph Databases," 29-Apr-2012.
[21] http://www.slideshare.net/maxdemarzi/introduction-to-graph-databases-12735789.
[22] "Thumbtack Technology - Ultra-High Performance NoSQL Benchmarking: Analyzing Durability and PerformanceTradeoffs."http://thumbtack.net/whitepapers/ultra-high-performance-nosql-benchmark.html.
[23] T. Rabl, S. Gómez-Villamor, M. Sadoghi, V. Muntés-Mulero, H.-A. Jacobsen, and S. Mankovskii, "Solving Big Data Challenges for Enterprise Application Performance Management," Proc VLDB Endow, vol. 5, no. 12, pp. 1724–1735, Aug. 2012.
[24] "Big Data Benchmark." https://amplab.cs.berkeley.edu/benchmark/.
[25] J. Han, M. Kamber, and J. Pei, Data mining: concepts and techniques. Burlington, MA: Elsevier, 2012.
[26] R. Tibshirani, "Glossary of Machine learning and statistical terms." Stanford University, 2012.
[27] B. Bederson and et. al., "Ordered and quantum treemaps: Making effective use of 2D space to display hierarchies."https://www.researchgate.net/publication/220184592_Ordered_and_quantum_treemaps_Making_effective_use_of_2D_space_to_display_hierarchies.
[28] "Force-directed graph drawing - Wikipedia, the free encyclopedia." http://en.wikipedia.org/wiki/Forcedirected_graph_drawing.
[29] B. Shneiderman, "Extreme Visualization: Squeezing a Billion Records into a Million Pixels."
[30] http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.145.2521.
[31] Z. Liu and et. al., "Stanford Vis Group | imMens: Real-time Visual Querying of Big Data."http://vis.stanford.edu/papers/immens
[32] R. Ostrovsky and et. al., "Private Searching On Streaming Data." http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.157.1127.
[33] S. Papadimitriou and et. al., "Time series compressibility and privacy."
[34] http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.66.7150.
[35] V. Karwa and et. al., "Private analysis of graph structure."
[36] http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.227.8815.
[37] D. Boneh and et. al., "Short Group Signatures." http://crypto.stanford.edu/~dabo/abstracts/groupsigs.html.
[38] P. Gaborit and et. al., "Lightweight code-based identification and signature."
[39] http://citeseer.uark.edu:8080/citeseerx/viewdoc/summary;jsessionid=410E9FDC4CC2BF40183250734B93BE7D?doi
[40] =10.1.1.131.3620.

[41] D. Boneh, B. Lynn, and H. Shacham, “Short Signatures from the Weil Pairing,” in Advances in Cryptology – ASIACRYPT
[42] 2001, C. Boyd, Ed. Springer Berlin Heidelberg, 2001, pp. 514–532.
[43] B. Przydatek and et. al., “SIA: secure information aggregation in sensor networks.”
[44] http://dl.acm.org/citation.cfm?id=958521.
[45] K. Kent and et. al., “Guide to Computer Security Log Management. SP 800-92.”
[46] http://dl.acm.org/citation.cfm?id=2206303.
[47] L. Sweeney, “k-anonymity: a model for protecting privacy.”
[48] http://dataprivacylab.org/dataprivacy/projects/kanonymity/kanonymity.html.
[49] A. Narayanan and et. al., “Robust De-anonymization of Large Sparse Datasets.”
[50] http://dl.acm.org/citation.cfm?id=1398064.
[51] B. Fung and et. al., “Privacy-Preserving Data Publishing: A Survey on Recent Developments.”
[52] http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.150.6812.
[53] C. Dwork, “Differential Privacy.” http://research.microsoft.com/apps/pubs/default.aspx?id=64346.
[54] P. Mohan and et. al., “GUPT: privacy preserving data analysis made easy.”
[55] http://dl.acm.org/citation.cfm?id=2213876.
[56] P. Domingos, “A few useful things to know about machine learning.” http://dl.acm.org/citation.cfm?id=2347755.
[57] “Spark | AMPLab – UC Berkeley,” AMPLab - UC Berkeley. https://amplab.cs.berkeley.edu/publication/.
[58] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, “Spark: Cluster Computing with Working Sets,”in Proceedings of the 2Nd USENIX Conference on Hot Topics in Cloud Computing, Berkeley, CA, USA, 2010, pp. 10–10.©