2. Big Data and Machine learning technologies

2.1. Understanding Big Data

Terminology Big Data appeared relatively recently. According to Google Trends it is noticeable that usage of this collocation dramatically rose from 2011. And terminology used almost everywhere and mostly it is misuse. It is extremely important to understand what Big Data is and how it is work before imbedding it into any production.

Practice shows that there are several definitions according to Big Data and it varies depending on where it is used. There some examples who generally people understand what Big Data is. Big Data – it is when data is over than certain amount of size, for example when data is more than 1 terabyte. Big Data – it is when it is not possible to handle data in one computer. There is another one, Big Data – it is any kind of data. Even there are many definitions about Big Data we can give one definition which is the most appropriate. Big Data is [data sets](https://en.wikipedia.org/wiki/Data_set) that are so voluminous and complex that traditional [data-processing](https://en.wikipedia.org/wiki/Data_processing) [application software](https://en.wikipedia.org/wiki/Application_software) inadequate to deal with them. Big data challenges include [capturing data](https://en.wikipedia.org/wiki/Automatic_identification_and_data_capture), [data storage](https://en.wikipedia.org/wiki/Computer_data_storage), [data analysis](https://en.wikipedia.org/wiki/Data_analysis), search, [sharing](https://en.wikipedia.org/wiki/Data_sharing), [transfer](https://en.wikipedia.org/wiki/Data_transmission), [visualization](https://en.wikipedia.org/wiki/Data_visualization), [querying,](https://en.wikipedia.org/wiki/Query_language) updating, [information privacy](https://en.wikipedia.org/wiki/Information_privacy) and data source. There are five concepts associated with big data: *volume*, *variety*, *velocity* and, the recently added, *veracity* and *value* [1].

Thereby, under the Big Data, we will not understand particular amount of data, or even the data itself, but the processing methods that allow them to process information. These methods can be applied to both huge data sets and to small datasets. To obtain those data sets we need some source that generates data sets. As a source of data set we can consider information about all transactions of bunk clients, information about purchases in huge stores and markets and so on. Amount of data sources increase swiftly. It means that technologies in handling data become in demand every day.

There is an opportunity for organizations to conduct more detailed and richer analysis of information about their customers if they are able to assemble the large amount of data about their clients. There are several ways to gather such kind of information. For example, it might be social network comments which are public shared, information gathered by volunteers using public questionnaires or some applications, electronic check-ins. In the era of smart devices and sensors data can be easily collected from broadband spectrum of circumstances.

To implement better analysis data should be well structured, but gathered data set is not always in the good form to be analyzed. Big Data can create some noise. If organizations need to collect the large amount of data it must be able to handle that collected data and at the same time determine which data represents sign of noise. The key factor is to ability to determine things that makes data more relevant. Before doing some actions upon data its format and structure might require certain handling. Because it becomes easy in storing and sorting structured data. Unstructured data often becomes useless.

Storing information according to Big Data mostly implemented on computer databases and there is specialized software to handle those large and very complex data sets. That large amount of data sets must be managed and to do so Software-as-a-Service (SaaS) organizations specialized in managing complex data sets. To find existence of correlation data analysts try to find relationship between different types of data.

One of the main understandings in terminology of Big Data is velocity. Velocity refers to how quickly big data can be analyzed. Data sources are correlated and analyzed by big data analysts try to give answer for some questions or give needed results using queries. All those activities need reliable velocity and moreover data analysts have to be able to understand what they are looking for.

The need for velocity in big data terminology leads to certain requirements for computer infrastructure to be corresponded. Organizations started to invest in computing power and create server cluster to faster process the large amount of data. There must be adequate computer power to solve Big Data tasks to retrieve valuable information.

Providing such velocity to analyze data can be a problem for enterprise in cost-effective manner. And many enterprise leaders are not ready to invest for such kind of computing power which can be used only to solve big data tasks. As a solution for this problem public cloud computing has emerged for hosting and processing data. Such cloud providers can host very large of data to accomplish big data projects for sure.

In the way of service level improvements some public cloud providers offer big data capabilities like Hadoop, data warehouses and other cloud services.

Data is nothing if it does not bring value. Value and effectiveness of data set directly depends on human operations on that data and formulating proper queries.

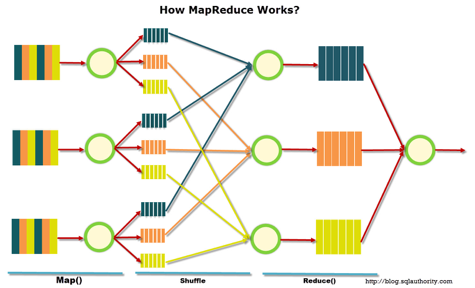
2.2 Principles of working with Big Data

According to the Big Data definition we can formulate the major principles of working with such data:

1. Horizontal scalability. Because there can be as much data as you like - any system that involves processing large data must be extensible. The amount of data increased twofold - the amount of iron in the cluster doubled and everything continued to work.
2. Fault tolerance. The principle of horizontal scalability implies that there can be many machines in a cluster. For example, Yahoo's Hadoop cluster has more than 42,000 machines (this link shows cluster sizes in different organizations). This means that some of these machines will be guaranteed to fail. Methods of working with large data must take into account the possibility of such failures and experience them without any significant consequences.
3. Locality of data. In large distributed systems, data is distributed over a large number of machines. If the data is physically located on one server and processed on another server, the data transmission costs may exceed the processing costs. Therefore, one of the most important principles for designing Big Data solutions is the principle of locality of data - if possible, we process data on the same machine on which we store it.

All modern working facilities with Big Data have to follow to those three principles. In order to follow them - it is necessary to invent some methods and paradigms for the development of data mining tools. One of the classical methods is Map Reduce.

Map Reduce is a distributed data processing model proposed by Google for processing large amounts of data on computer clusters. In the following illustration we can see how Map Reduce works [2]:



Picture 1.

MapReduce assumes that the data is organized in the form of some records. Data processing takes place in 3 stages:

The first is the Stage Map. At this stage, the data is preprocessed using the map () function, which is defined by the user. The work of this stage is to preprocess and filter the data. The operation is very similar to the map operation in functional programming languages ​​- the user-defined function is applied to each input record.

The map () function applied to one input record and produces a lot of key-value pairs. The set - i.e. can issue only one record, cannot produce anything, but can give out several key-value pairs. What will be in the key and in the meaning is to be solved by the user, but the key is very important, since data with one key will fall into one instance of the reduce function in the future.

The second one is Stage Shuffle. Passes unnoticed for the user. At this stage, the output of the map function is "sorted by baskets" - each basket corresponds to one output key of the map stage. In the future, these baskets serve as an input for reduce.

Finally it is the Reduce step. Each "basket" with values ​​generated at the shuffle stage falls into the input of the reduce () function.

The reduce function is set by the user and computes the final result for a separate "basket". The set of all values ​​returned by the reduce () function is the final result of the Map Reduce task.

A few additional facts about Map Reduce:

All the starts of the map function work independently and can work in parallel, including on different machines of the cluster.

All executions of reduce function work independently and can work in parallel, including on different machines of the cluster.

Shuffle within itself represents a parallel sort, so it can also work on different cluster machines. Items 1-3 allow the principle of horizontal scalability to be fulfilled.

The map function, as a rule, is used on the same machine, on which the data is stored - this allows to reduce data transmission over the network (the principle of locality of data).

Map Reduce is always a full scan of the data, there are no indexes. This means that Map Reduce is badly applied when the response is required very quickly.

2.3 Understanding Machine Learning