EMERGING TRENDS AND TECHNOLOGIES IN BIG DATA ANALYTICS: A COMPRENSIVE REVIEW

Prof.Sanchita.N.Nawale[1], Prof.B.N.Chaudhari[2], Prof.S.B.Phatangare[3]

[1] *Assistant Professor, Computer Engineering Department, AVCOE, Sangamner, Maharashtra, India*
[2] *Assistant Professor, Computer Engineering Department, AVCOE, Sangamner, Maharashtra, India* [3]
Assistant Professor, Computer Engineering Department, AVCOE, Sangamner, Maharashtra, India

ABSTRACT

Today Big Data draws a lot of attention in the IT world. The rapid rise of the Internet and the digital economy has fuelled an exponential growth in demand for data storage and analytics, and IT department are facing tremendous challenge in protecting and analyzing these increased volumes of information. The reason organizations are collecting and storing more data than ever before is because their business depends on it. The type of information being created is no more traditional database-driven data referred to as structured data rather it is data that include documents, images, audio, video, and social media contents known as unstructured data or Big Data. Big Data Analytics is a way of extracting value from these huge volumes of information, and it drives new market opportunities and maximizes customer retention. After reviewing the methodology of research used in creating the review the investigation of big data will begin by attempting to craft a satisfying definition of the term. Many of the relevant technologies and techniques used in big data analytics will be covered briefly and the benefits of big data analytics across various sectors will be explored. The review will also present several of the challenges and barriers faced by purveyors of big data analytic tools and attempt to determine if the results of the analytics offset the costs of overcoming these challenges sufficiently to make them a wise investment.

Keyword : - *Big Data, Analytics, Hadoop, MapReduce*

1. INTRODUCTION

Big Data is an important concept, which is applied to data, which does not conform to the normal structure of the traditional database. Big Data consists of different types of key technologies like Hadoop, HDFS, NoSQL, MapReduce, MongoDB, Cassandra, PIG, HIVE, and HBASE that work together to achieve the end goal like extracting value from data that would be previously considered dead. According to a recent market report published by Transparency Market Research, the total value of big data was estimated at $6.3 billion as of 2012, but by 2018, it's expected to reach the staggering level of $48.3 billion that's almost a 700 percent increase [29]. Forrester Research estimates that organizations effectively utilize less than 5 percent of their available data. Big Data Analytics reflect the challenges of data that are too vast, too unstructured, and too fast moving to be managed by traditional methods. From businesses and research institutions to governments, organizations now routinely generate data of unprecedented scope and complexity. Gleaning meaningful information and competitive advantages from massive amounts of data has become increasingly important to organizations globally. Trying to efficiently extract the meaningful insights from such data sources quickly and easily is challenging. Thus, analytics has become inextricably vital to realize the full value of Big Data to improve their business performance and increase their market share.

There are several big data platforms available with different characteristics and choosing the right platform requires an in-depth knowledge about the capabilities of all these platforms [1]. Especially, the ability of the platform to adapt to increased data processing demands plays a critical role in deciding if it is appropriate to build the ana- lytics based solutions on a particular platform.

BIG DATA TECHNOLOGIES

Apache Flume

Apache Flume is a distributed, reliable, and available system for efficiently collecting, aggregating and moving large amounts of log data from many different sources to a centralized data store. Flume deploys as one or more agents, each contained within its own instance of the Java Virtual Machine (JVM). Agents consist of three pluggable components: sources, sinks, and channels. Flume agents ingest incoming streaming data from one or more sources. Data ingested by a Flume agent is passed to a sink, which is most commonly a distributed file system like Hadoop. Multiple Flume agents can be connected together for more complex workflows by configuring the source of one agent to be the sink of another. Flume sources listen and consume events. Events can range from newline-terminated strings in stdout to HTTP POSTs and RPC calls — it all depends on what sources the agent is configured to use. For example, if the network between a Flume agent and a Hadoop cluster goes down, the channel will keep all events queued until the sink can correctly write to the cluster and close its transactions with the channel. Sink is an interface implementation that can remove events from a channel and transmit them to the next agent in the flow, or to the event's final destination and also sinks can remove events from the channel in transactions and write them to output. Transactions close when the event is successfully written, ensuring that all events are committed to their final destination.

Apache Sqoop

Apache Sqoop is a CLI tool designed to transfer data between Hadoop and relational databases. Sqoop can import data from an RDBMS such as MySQL or Oracle Database into HDFS and then export the data back after data has been transformed using MapReduce. Sqoop also has the ability to import data into HBase and Hive. Sqoop connects to an RDBMS through its JDBC connector and relies on the RDBMS to describe the database schema for data to be imported. Both import and export utilize MapReduce, which provides parallel operation as well as fault tolerance.

Apache Pig

Apache's Pig is a major project, which is lying on top of Hadoop, and provides higher-level language to use Hadoop's MapReduce library. Pig provides the scripting language to describe operations like the reading, filtering and transforming, joining, and writing data which are exactly the same operations that MapReduce was originally designed for. Instead of expressing these operations in thousands of lines of Java code which uses MapReduce directly, Apache Pig lets the users express them in a language that is not unlike a bash or Perl script.Pig was initially developed at Yahoo Research around 2006 but moved into the Apache Software Foundation in 2007. Unlike SQL, Pig does not require that the data must have a schema, so it is well suited to process the unstructured data. But, Pig can still leverage the value of a schema if you want to supply one. PigLatin is relationally complete like SQL, which means it is at least as powerful as a relational algebra.

Apache Hive

Hive is a technology developed by Facebook that turns Hadoop into a data warehouse complete with a dialect of SQL for querying. Being a SQL dialect, HIVEQL is a declarative language. In PigLatin, you specify the data flow, but in Hive we describe the result we want and hive figures out how to build a data flow to achieve that result. Unlike Pig, in Hive a schema is required, but you are not limited to only one schema.

Apache ZooKeeper

Apache Zoo Keeper is an effort to develop and maintain an open-source server, which enables highly reliable distributed coordination. It provides a distributed configuration service, a synchronization service and a naming registry for distributed systems. Distributed applications use ZooKeeper to store and mediate updates to import configuration information. ZooKeeper is especially fast with workloads where reads to the data are more common than writes.

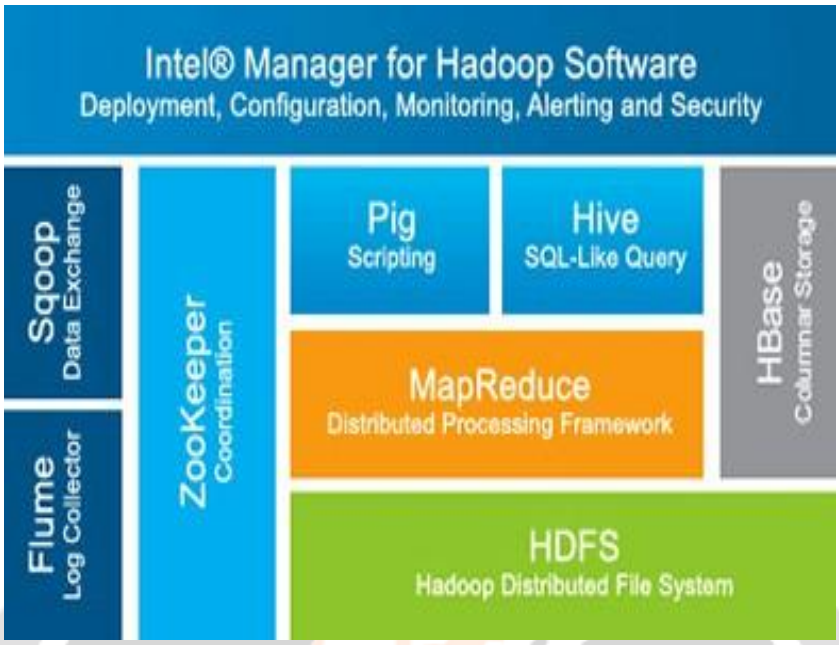

Fig -1: Big Data Technologies

MongoDB

MongoDB is an open source, document-oriented NoSQL database that has lately attained some space in the data industry. It is considered as one of the most popular NoSQL databases, competing today and favors master-slave replication. The role of master is to perform reads and writes whereas the slave confines to copy the data received from master, to perform the read operation, and backup the data. The slaves do not participate in write operations but may select an alternate master in case of the current master failure. MongoDB uses binary format of JSON-like documents underneath and believes in dynamic schemas, unlike the traditional relational databases. The query system of MongoDB can return particular fields and query set compass search by fields, range queries, regular expression search, etc. and may include the user-defined complex JavaScript functions.

Apache Hadoop

The Apache Hadoop software library is a framework that enables the distributed processing of large data sets across clusters of computers. It is designed to scale up from single servers to thousands of machines, with each offering local computation and storage. The basic notion is to allow a single query to find and collect results from all the cluster members, and this model is clearly suitable for Google's model of search support. One of the largest technological challenges in software systems research today is to provide mechanisms for storage, manipulation, and information retrieval on large amount of data. Web services and social media produce together an impressive amount of data, reaching the scale of petabytes daily (Facebook, 2012). These data may contain valuable information, which sometimes is not properly explored by existing systemsHadoop is a popular choice when you need to filter, sort, or pre-process large amounts of new data in place and distill it to generate denser data that theoretically contains more information. Pre-processing involves filtering new data sources to make them suitable for additional analysis in a data warehouse. Hadoop is a top-level open source project of the Apache Software Foundation. Several suppliers, including

Intel, offer their own commercial Hadoop distributions, packaging the basic software stack with other Hadoop software projects such as Apache Hive, Apache Pig, and Apache Sqoop.

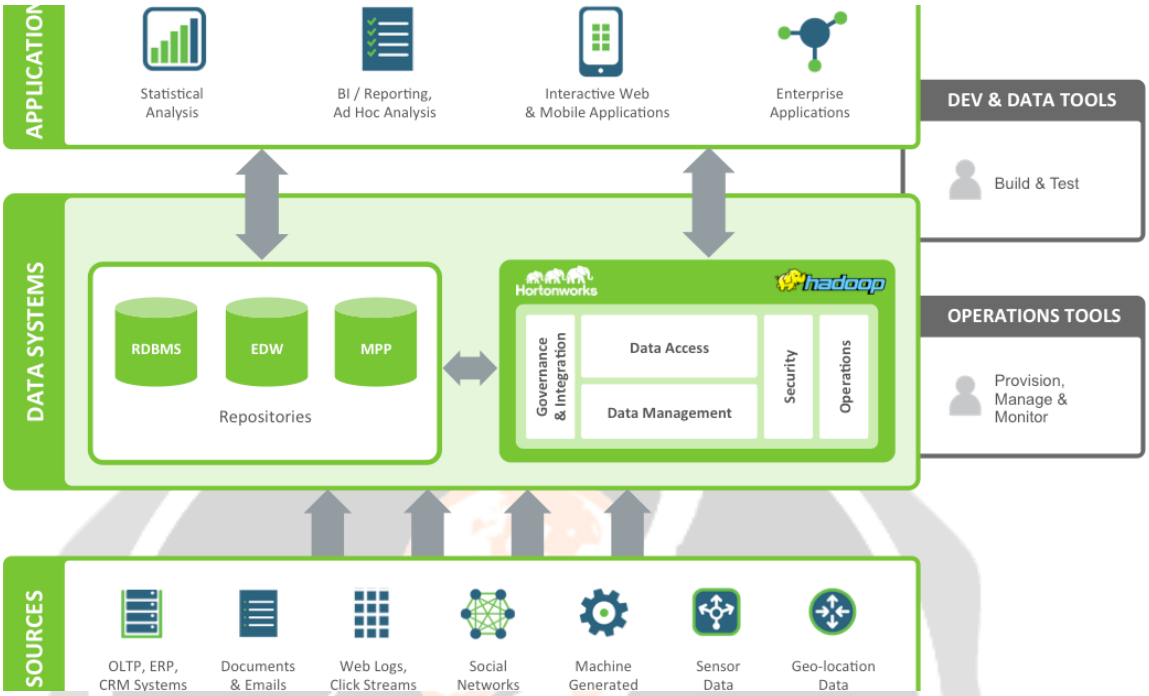

Fig -1: Big Data Framework

MapReduce

MapReduce is the model of distributed data processing introduced by Google in 2004. The fundamental concept of MapReduce is to divide problems into two parts: a map function that processes source data into sufficient statistics and a reduce function that merges all sufficient statistics into a final answer. By definition, any number of concurrent map functions can be run at the same time without intercommunication. Once all the data has had the map function applied to it, the reduce function can be run to combine the results of the map phases. For large scale batch processing and high speed data retrieval, common in Web search scenarios, MapReduce provides the fastest, most cost-effective and most scalable mechanism for returning results.

Techniques

There are a myriad of analytic techniques that could be employed when attacking a big data project. Which ones are used depends on the type of data being analyzed, the technology available to you, and the research questions you are trying to solve. Some of the tools that came up frequently in the reviewed material are summarized here.

- EDWs: Enterprise data warehouses are databases used in data analysis. Russom (2011) writes that for many businesses that are trying to start handling big data the big question is “Can the current or planned enterprise data warehouse (EDW) handle big data and advanced analytics without degrading performance of other workloads for reporting and online analytic processing?” Some institutions manage their analytic data in the EDW itself while others use a separate platform, which helps relieve some of the stress on the server resulting from managing your data on the EDW.

- Visualization products: One of the difficulties with big data analytics is finding ways to visually represent results. Many new visualization products aim to fill this need, devising methods for representing data points numbering up into the millions. Russom (2011) lists this field as one of those having the most potential, saying it is "poised for aggressive adoption." Beyond simple representation visualization can also help in the information search.
- NoSQL databases: These databases are designed specifically to deal with very large amounts of information that don't utilize a relational model.
- Text analytics: A large portion of generated data is in text form. Emails, internet searches, web page content, corporate documents, etc. are all largely text based and can be good sources of information. Text analysis can be used to extract information from large amounts of textual data.

2.BIG DATA FRAMEWORK

Apache Spark

Apache Spark an open source big data processing framework built around speed, ease of use, and sophisticated analytics. It was originally developed in 2009 in UC Berkeley's AMP Lab, and open sourced in 2010 as an Apache project. Hadoop as a big data processing technology has been around for ten years and has proven to be the solution of choice for processing large data sets. MapReduce is a great solution for one-pass computations, but not very efficient for use cases that require multi-pass computations and algorithms. Each step in the data processing workflow has one Map phase and one Reduce phase and you'll need to convert any use case into MapReduce pattern to leverage this solution. Spark takes MapReduce to the next level with less expensive shuffles in the data processing.

Spark also supports lazy evaluation of big data queries, which helps with optimization of the steps in data processing workflows. It provides a higher-level API to improve developer productivity and a consistent architect model for big data solutions. Spark holds intermediate results in memory rather than writing them to disk, which is very useful especially when you need to work on the same dataset multiple times. It's designed to be an execution engine that works both in-memory and on-disk. Spark operators perform external operations when data does not fit in memory. Spark can be used for processing datasets that larger than the aggregate memory in a cluster. Spark will attempt to store as much as data in memory and then will spill to disk.

Berkeley data analytics stack (BDAS)

The Spark developers have also proposed an entire data processing stack called Berkeley Data Analytics Stack (BDAS) [20] which is shown in Figure 2. At the lowest level of this stack, there is a component called Tachyon [21] which is based on HDFS. It is a fault tolerant distributed file system which enables file sharing at memory-speed (data I/O speed comparable to system memory) across a cluster. It works with cluster frameworks such as Spark and MapReduce. The major advantage of Tachyon over Hadoop HDFS is its high performance which is achieved by using memory more ag- gressively. Tachyon can detect the frequently read files and cache them in memory thus minimizing the disk access by different jobs/queries. This enables the cached files to be read at memory spee.

The second component in BDAS, which is the layer above Tachyon, is called Apache

Mesos. Mesos is a cluster manager that provides efficient resource isolation and sharing across distributed applications/frameworks. It supports Hadoop, Spark, Aurora [22], and other applications on a dynamically shared pool of resources. With Mesos, scalability can be increased to tens of thousands of nodes. APIs are available in java, python and C++ for developing new parallel applications. It also includes multi-resource scheduling capabilities.

Recently, BDAS and Spark have been receiving a lot of attention due to their per- formance gain over Hadoop. Now, it is even possible to run Spark on Amazon Elastic Map-Reduce [26]. Although BDAS consists of many useful components in the top layer (for various applications), many of them are still in the early stages of development and hence the support is rather limited. Due to the vast number of tools that are already available for Hadoop MapReduce, it is still the most widely used distributed data pro- cessing framework

REFERENCES

1. Apache Software Foundation. (2010). Apache ZooKeeper. Retrieved April 5, 2015 from https://zookeeper.apache.org
2. Chae, B., Sheu, C., Yang, C. and Olson, D. (2014). The impact of advanced analytics and data accuracy on operational performance: A contingent resource based theory (RBT) perspective, Decision Support Systems, 59, 119-126.
3. Chambers, C., Raniwala, A., Adams, S., Henry, R., Bradshaw, R., and Weizenbaum, N. (2010). Flume Java: Easy, Efficient Data-Parallel Pipelines. Google, Inc. Retrieved April 1, 2015 from http://pages.cs.wisc.edu/~akella/CS838/F12/838-CloudPapers/FlumeJava.pdf
4. Cisco Systems. Cisco UCS Common Platform Architecture Version 2 (CPA v2) for Big Data with Comprehensive Data Protection using Intel Distribution for Apache Hadoop. Retrieved March 15, 2015, from http://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/UCS_CVDs/Cisco_UCS_CPA_for_Big_Data_wi th_Intel.html
5. Miller, K. (n.d.). Big Data Analytics in Biomedical Research. Biomedical Computation Review, (Winter 2011/2012), 14–21.
6. Picciano, A. G. (2012). The Evolution of Big Data and Learning Analytics in American Higher Education. Journal of Asynchronous Learning Networks, 16(3), 9– 20.