Abstract: Data Analytics is the trending domain that analyses data to observe patterns and predict future outcomes. The outcomes are based upon analysis of past and current trends and behaviors. Data analytics deals with both descriptive and predictive analyses of data. Descriptive Data Analytics summarizes the data, it's behavior and draws useful conclusion from it. Predictive Data Analytics is the branch of data analytics that predicts future outcomes based on the current and historical data. These future predictions are drawn by observing patterns followed for past data and outcomes for the past events for similar scenarios. In this paper, various branches of data analytics have been discussed. Big data analytics architecture gives an overview of the various tools and system structure involved in big data analytics. Big data analytics is closely related to data mining and hence, implements data mining algorithms. Latter part of the paper covers machine learning algorithms and neural networks for training the dataset to recognize patterns for the modeled data and predict outcomes based on the training and pattern recognition. Modeling of data using neural networks helps in generating accurate and exhaustive outcomes.

Keywords: big data, cloud, analytics, regression, neural, clustering, classification

I. INTRODUCTION

Big Data deals with the process of collection, processing and querying of the essential data that is always in larger quantities (exabytes or more). Big Data is always seen in terms of its volume, velocity, variety, veracity and value to the business process. It is mostly subjected in terms of large processing systems such as grids but majority of its aspect is shifted to cloud systems [1] where it totally satisfies the economic, technical, functional and operational feasibility of the organizations working on big data. The consumerization of the IT sector mainly focuses on mobility, mobile services and mobile applications as the underlying foundation technology which is the backbone gets upgraded with each requirement as needed.

II. CLOUD SYSTEMS

A cloud system is a combination of many diverse distinct complex systems, infrastructures, platforms and software which provide a set of services and functionalities from a shared pool of resources. Data in cloud system is always distributed in nature but appears to be one and a complex level of transparencies is involved. Cloud platform is always in demand and mostly used by commercial and scientific communities for analytics purposes. There is a constant requirement for intelligent and scalable analytics methods, tools and applications by the consumers. These intelligent analytics need to be implemented on complex, diverse or real time data. The three commonly used cloud platforms to implement big data analytics solutions are PaaS (Platform as service), IaaS(Infrastructure as service) and SaaS(Software as service) [2]. The IaaS platform is generally used by data managers and researchers, the SaaS platform is commonly used by software programmers and analysts and the PaaS is widely used by data scientists and data mining application developers. Cloud based big data systems use different deployment schematics [3] based upon the processing and result generation requirements; these are mostly subjected to:

- **Adhoc Based**: This is followed when analytics is to be performed on data that is new or collected from various distributed sources on demand basis. For this the research areas are open in the fields of the catalog of migration patterns and predictive resource scaling for cloud platforms.

- **Batch Based**: This process is constant and involves long continuous analytics operations on data which is frequently updated in the real time or an offline environment, collected from various by means of ETL cycles. Such schematics are generally subjected to prolong service lifecycle management and maintenance.

![Fig. 1. Cloud Computing Data Processing Architecture](image)
A cloud System is a modern revolutionary platform that provides a lot of growth opportunities with a lot of business benefits. Currently there are lots of research areas available in terms of cloud systems and computing [3]. One of the biggest research challenges lie in the convergence of various technologies and resources and placing it in terms of cloud for the system to appear as whole and compatible. When big data is subjected to cloud systems and computing aspects, the scalability and multi-tenancy of the data along with the placement optimization algorithms of data in storages are always the point of interests.

- **Cloud Platform Management**: The challenges for this aspect lie in implementing effective middleware services for designing, deploying, integrating and managing big data applications in a diverse, elastic and scalable environment. The scalability and multi-tenancy of the data requires the implementation of various optimization algorithms of data in storages.

- **Cloud Enabled Big Data Applications and Platforms**: Such entities are supposed to take optimal advantage of the scalability, processing and reliability of the cloud systems.

- **Big Data Aggregation**: Aggregation of large amount of data from diverse cloud/grid or other relevant network storages with the support of additional layers of service management is a complex and time consuming process. Development of optimal techniques and methods to increase the performance of this activity are always in the design and research phase.

- **Big Data Interoperability for Cloud Systems**: Challenges to ensure that the cloud services performing and providing big data operations(functionalities) work together and interoperate successfully. This also includes ensuring common and standard interfaces for cloud computing along with the portability of cloud virtualization across diverse underlying technologies.

Data interoperability and data compatibility [4] is a big challenge for the big data analytics operations. The data which is in structured or unstructured format needs to be integrated from various sources. After this, the only required amount of data is to be targeted and analyzed. The main challenges lie in the types of data storages to be used, diverse nature of the data, structures and levels of data integration and operations that are to be carried out. The data stored in such systems could be in form of images, videos, documents or any other types and is in the size of Exabytes.

Consider a scenario where there is an outbreak of influenza in a particular region and there is an immediate requirement to analyze the impact of this outbreak on the people as well as the areas of its impact. The data gathering process needs to target the health ministry department databases for the registered cases, geographical area information as well as information from social media such as Facebook, Twitter etc. for its real time analysis. The data in this case needs to be gathered from structured data source like RDBMS and also from a real time data sources and proper analytics algorithm needs to be applied for the efficient solutions. When the primary data source was integrated with the data from various social media sources, it provides a very high level precision and increases accuracy in the results. So the data has to be made compatible in the most optimized way by extracting only the required co-related and inter related information from the targeted data sources as per the analytics requirement. This also leads to the design and

---

**III. DATA ANALYTICS AND INTEROPERABILITY**

Data interoperability and data compatibility [4] is a big challenge for the big data analytics operations. The data which is in structured or unstructured format needs to be integrated from various sources. After this, the only required amount of data is to be targeted and analyzed. The main challenges lie in the types of data storages to be used, diverse nature of the data, structures and levels of data integration and operations that are to be carried out. The data stored in such systems could be in form of images, videos, documents or any other types and is in the size of Exabytes.

Consider a scenario where there is an outbreak of influenza in a particular region and there is an immediate requirement to analyze the impact of this outbreak on the people as well as the areas of its impact. The data gathering process needs to target the health ministry department databases for the registered cases, geographical area information as well as information from social media such as Facebook, Twitter etc. for its real time analysis. The data in this case needs to be gathered from structured data source like RDBMS and also from a real time data sources and proper analytics algorithm needs to be applied for the efficient solutions. When the primary data source was integrated with the data from various social media sources, it provides a very high level precision and increases accuracy in the results. So the data has to be made compatible in the most optimized way by extracting only the required co-related and inter related information from the targeted data sources as per the analytics requirement. This also leads to the design and
development of an effective and much more optimized ETL (Extract Transform and Load) cycle [5].

Since various data sources have data in different types data, the challenge lies in extracting this data especially when the data generation with different pattern process is larger and when a lot of complexities involved in case of real time unstructured data generation. This data is to be optimized further before the next stage and the requirement of resources for this task is much more. After the extraction process, the data needs to be transformed and clubbed into a common compatible platform, then this data is made available to the respective analytics engines as per the application necessity. The particular analytics engine makes the desired request of data as required and hence the type of extract and transform cycle operations are dependent on it. There is a parallel access to the data simultaneously so the rules and constraints of distributed access have to be imposed. Analytics engine operations are stream processing based or batch processing based and sometime require the combination of both as per the demands. In stream based processing the latest recently acquired data is analyzed on demand basis and results are produced. The classic example of this type of processing is network traffic monitoring. Whereas in batch based processing the large amount of data is divided into smaller portions and then these portions are then processed in a distributed manner, analytics on these portions run in parallel and the required results are combined as needed. This type of processing is commonly used in the fields of bioinformatics and web mining. The queries design and optimization process [6] is much complicated and the correct methods to execute the queries have to be implemented at various stages. The overall data management factor is very crucial in big data compatibility or interoperability, the interfacing functions among various participating entities, intelligent querying techniques and application of the right programming models plays a vital role in its successful operations. The scalability factor has to be considered when the integration requirement demand with various new data sources keep on increasing. The factors such as architectural restrictions, response time, latency issues, utilization and performance of various computing resources and diverse storage devices/networks are to be considered as

the system scalability increases. As highlighted in [7] the other two factors such as hardware and support systems are to be considered when analyzing the big data interoperability issues. The storage systems for the data sources need to be very reliable and should provide rapid access with the assurance of very less or no latency. The availability of secondary support systems is very essential benefical for the above mentioned process. These support systems can provide with the functionalities of fault tolerance, data backups, system monitoring, load balancing, rapid access and control, error reporting and handling.

IV. DATA ANALYSIS

A. Clustering

These algorithms are used to sort data into clusters based on certain metrics that assays how close a particular data item is to another data item based on the said metric. There are multiple types of clustering algorithms and they can broadly be divided into the following categories:

- **Partition based**: All clusters are predetermined. The algorithms allocate data objects to various partitions where each partition is treated as a cluster. The data objects within the partition are closely related with other data objects from the same cluster as compared to data objects from other clusters in terms of the predefined metrics. The groups can initially be made arbitrarily following which, the data objects are reallocated to the most suitable partitions. The two basic requirements for a partition to be treated as a cluster are: A group should have at least one data object and a data object should be part of only one group. Examples: PAM, CLARA, FCM, etc.

- **Hierarchical based**: Data objects are grouped together in a hierarchical fashion based on the proximity of each data object. Every proximity is generated using second tier nodes from the dataset. The hierarchical method of clustering is of two types: Bottom-up and top-bottom. In the bottom-up algorithm, each data object is initially considered as a unique cluster and multiple clusters join to form new, larger clusters in a recursive manner. In the top-bottom algorithm, the entire dataset is initially considered as a single cluster and chunks of data split up to form smaller clusters, repeatedly. In both the types, an exit condition is required to stop the recursion. A major drawback of hierarchical algorithms is that tracking back a merge or a split is not possible hence no step can be undone. Examples: BIRCH, Chameleon, ROCK, etc.

- **Density based**: These algorithms are used to separate data objects based on their density, boundaries and connectivity. These algorithms are very similar to point nearest neighbors. The clusters that are formed grow in any direction, absorbing new data objects randomly in any path that the density takes. The new clusters formed take an arbitrary shape and size and are usually safe against outliers. To find the methods that affect any one data point, the overall density of a point is analyzed. Examples:
OPTICS, DBSCAN, DENCLUE, etc. These are all mainly used to filter the noise (i.e. the outliers) from a given dataset.

- **Grid based:** These algorithms are used for their speed of execution. This approach greatly cuts down processing time by dividing up the dataset into grids. The algorithm combs the entire dataset the first time to find the statistical values of the grid elements. This technique gives the values to the grid that renders the technique independent of the size of the dataset. A uniform grid is used to obtain regional data which is then used for clustering rather than the traditional method of working directly on the database. The performance of these algorithms is solely based on the size of the grids. The problem arises when the database has highly irregular data objects. In this case, a solitary grid may prove to be insufficient for quickly clustering the data objects or satisfying the time constraints. Examples: STING

- **Model based:** It is based on the presumption that the data objects of the database are generated by probability distributions. Optimization of the fit or the characteristics of the data and some already existing mathematical models takes place. The predefined models help to automate the process of calculating the number of clusters and each of their sizes. This is done after taking into account statistics, noise, etc. There are two types of methods: statistical and neural. The statistical way employs probability to find the clusters and the neural technique employs a set of IO nodes that have weights associated with them. Neural networks are usually more complicated to implement but have a better efficiency and hence preferred to the statistical approach. Further, neural networks are known to give more accurate clusters. Examples: MCLUST, COBWEB, etc.

### B. Classification

Unlike clustering, groups are predefined when it comes to classification with each data instance belonging to a particular class. The grouping is not done with the intra-relativity of the data items but rather with respect to the preset class parameters.

- **Decision tree:** A decision tree is a flowchart in the shape of a tree consisting of rectangles (internal nodes), elliptical leaves (Internal representation) and circles (Leaf nodes). Each node is a test attribute, each branch is an output test and each leaf is decision class. This is the most basic algorithm of classification. It compares the input data to the various classes and provides a decision of which class the input data falls under. For example, to classify a customer as a high valued, medium valued or low valued customer, the algorithm will take in to account various factor any of which could shift the customer from a class that he/she seemed obvious to be a part of to any other class that was not initially expected.

- **Tree pruning:** This is an extension to the decision tree method. When a large amount of data has to be worked upon, the decision tree could end up becoming too big to traverse at a desirable speed. However, it can be noticed that a high number of the branches of the tree could be unreliable, irrelevant or outright useless while classifying different types of data. This problem can be solved by pruning or reducing the size of the tree by removing some of the branches – the most unreliable of all. Pruning and post pruning are the two types of pruning methods used to improve speed of the tree.

- **Bias:** This involves neural networks and certain amount of fuzzy logic. During the training phase of the neural network, it adapts certain bias and weights. The idea is that because the classification is fuzzy, a data set may belong to more than one class. The algorithm has to check the level of belongingness to a particular class and minimize the error term.

### C. Predictive Data Mining

It is a subset of data mining which involves the prediction of probabilities, statistics and trends observed in the future. At the center of this entire system is a variable called a predictor which is measured using various algorithms and techniques against patterns and past data for individual objects or entire datasets to predict the future behavior of said data. A dataset could have multiple such variables each of whose results can be combined to come up with an overall conclusion (prediction). The basic methodology involves collecting data, finding patterns and making predictions to form a model. This model is tested and modified as more data becomes available to provide accurate conclusions with maximum reliability. As the model keeps getting modified, the accuracy increases. Some of the analysis techniques are:

- **Linear Regression:** [9] Also known as straight line regression, it is the most basic approach to predictive data mining. The predictor function i.e. the function used to determine the predictor variable is linear and the unknown parameters are calculated from the dataset. The basic function is in terms of X and Y where Y is the function to be calculated when X changes by unity. $Y = b + wX$

  As seen, Y is a straight line function where b and w are coefficients of regression and can be assumed to be weights. Here, X is the predictor variable.

- **Multiple Linear Regression:** [10] In linear regression, the function contained a single predictor variable. Here, the number of variables is more than one. The idea is that there are many independent variables on which one dependent variable depends to obtain the value. Here the dependent variable is Y while the independent variables are $X_i$ where $i > 1$. The equation is as follows: $Y = a + \sum b_k X_k + e$.

  As seen, there are multiple values of X where $a, b_k$ are the parameters (equivalent to b and w, respectively in linear regression). In short, MLR is the combination of correlation and linear regression which involves the statistical measurement of a single dependent on multiple independents.
• **Logistic Regression:** [11][12] The predictive model consists of target variables that can have exactly one of two values. For example: black/white, odd/even, win/lose, one/zero, etc. This form of regression is used for binomial regression. Apart from this, it can also be used for continuous target variables that model the probability of occurrence as a straight line function of a dataset of predictor variables. The equation: 

$$F(Z) = 1 / (1 + e^{-Z}) \ Z$$

is called the logit and it can take any real value while the output remains 0 or 1. Here, 

$$Z = b_0 + \sum b_i x_i$$

where b are the regression coefficients.

• **Poisson Regression:** [13] This form of regression where the dependent is a counter that restricts the previous regression w.r.t a binomial distribution. This means that the probability of success is relatively small because the total trials are very high. Often, it can be called the log linear model provided: it assumes that Y is over a Poisson distribution and can take a log of the expected value, characterized by a straight combination of unknowns.

$$\log (E(Y|x)) = a + bx$$

Where a and b are estimated by maximum possibility.

• **Regression Tree:** [14][15] Sometimes assumed to be a variation of decision trees, regression trees can make approximations to real valued functions. The tree is created by recursive partitioning of the data by splitting it at each level and each branch. The final output is a numerical variable, obtained by working on a combination of continuous variables. The algorithm picks a point at which the data is split in such a way that the sum of the deviation (squared) from the mean is minimized in both parts of the divided training data. The method is easy to understand and does not involve any difficult calculations. A glance at the tree will tell the observer which variable is making the prediction. If by chance the data is missing, one may not be able to continue traversal in the tree but a calculated prediction can be made by taking the mean of all the leaf nodes that are possible to be reached.

V. **MACHINE LEARNING AND NEURAL NETWORKS**

A. **System Architecture**

A specific software architecture is needed for involves real-time analytics.

1) **Batch Machine Learning Module:** This module is appropriate for tasks that are not time critical. It performs the following tasks:

a) **Query Analytics:** Information is extracted from the raw data. These analytics include computing data distributions operating over subsets of the data, mapping data to a different domain and generating data summaries to be passed to the dashboard for its visualization.

b) **Performing Data clustering:** Data involves several records and contain a specified distance function among them. These records can be grouped in several similar clusters.

c) **Building machine learning models, in order to enable fast processing later by the stream machine learning module[16].** For example naive Bayes classifiers, bayesian networks, neural networks, decision trees, markov models, etc.

2) **Stream Processing Module:** Designed for time-critical applications, provides the results in fraction of seconds. It applies various machine learning techniques along with real-time stream of data. It adapts well to perform following tasks:

a) **Prediction:** Making prediction over future-based event based on historical events.

b) **Classification and Segmentation:** Assign labels or predefined classes to an instance given set of classes or types.

c) **Recommendation:** Suggests a product to a customer based on his potential interest.

3) **Storage Module:** It is based on the Hadoop Distributed File System, that stores raw data such as historical data or text files. HDFS provides reliable distributed storage over cluster of computers by managing data distribution and replication over each node. For storing structured or semi-structured information, Apache HBase is a reliable option as it is a non-relational database built on top of HDFS. Thus, raw data is used for batch processing and for retrieving information stored in HBase [18].

4) **Dashboard:** It provides graphical way of visualising analytics performed over data generated from Batch Processing Module. It displays useful results which reveals relevant insights into the data. It also provides feedback to the user regarding the data.

5) **RESTful API:** It offers real-time analysis tools as a service for data analytics. All the tools that are important to perform analytics automatically stream the data to provide the advantage from real-time features.

B. **Genetic Algorithms**

Genetic algorithms are algorithms for optimization and machine learning based loosely on several features of biological evolution. The basic operators for Genetic Algorithm are crossover, selection and mutation [17].

- **Selection**—“Survival of the fittest”. Preference given to better outcomes

- **Mutation**—Randomly-trying and evaluating success to failure ratios.

- **Crossover**—or combining portions of good outcomes in the hope of creating an even better outcome.

C. **Self Organised and Layered Approach**

- **Self-Organizing Map:** Neurons are configured according to the topological structure of the input data. Iterative tuning of weights of neurons to approximate input data.
They are used to classify real-time transactions into genuine and fraudulent transactions.

- **Layered Approach**: Multi-layered approach consisting of initial layer (authentication and screening layers), core layer (risk scoring and behavior analysis layer) and final layer (review and decision making layer). The Core layer consists of two sub-layers: SOM layer followed by either a feed-forward neural network or a rule-based risk scoring systems.

**D. Probabilistic neuro-adaptive approach for data mining**

This section deals with the probabilistic Neuro-adaptive approach for data mining which gives us success to failure ratios in data mining for given set of data. These approaches are useful in detection of frauds in real-time transactions.

1) **Modeling the data:**
   a) Probability of risk or incorrect data is very low.
   b) The confidence limit for a transaction abort is very subjective and subject to client policy.
   c) Transactions with a confidence of fraud of higher than 10% are accepted to be revised or aborted- “Questionable Transaction. needs to be reviewed”.

2) **Mining the symbolic data:**
   a) Transaction is a data tuple with features xi: (x1,x2,x3,x4…..xn)
   b) All misuse transactions can be seen as kind of RULE: If symbolic features are given, then misuse occurs.

3) **Generalizing and weighting the association rules:**
   a) It aims at shortening raw associations by generalizing them to the most common transactions.
   b) For generalization, start comparing each transaction with the fraud transaction database to find pairs of similar transactions.
   c) Each pair is then merged into a generalized rule by replacing a non-identical feature by a ‘don’t care’-symbol ‘*’.
   d) The share of a fraud rule is defined as the percentage of fraud transactions covered by the rule.
   e) We define the confidence level in fraud diagnosis as: Confidence = Number of misuse covered by the rule / Number of transactions covered by the rule

**VI. CONCLUSION**

The research opportunities in the field of big data are tremendous. Some of the systems mentioned are already in use and new methods of optimizing their performance are always being looked for. Over the years, big data has changed in many ways and the traditional algorithms have been improved.

This area of research is now interlinked with many other fields such as neural networks, genetic algorithms and artificial intelligence. The technology is ever changing and as the amount of data to be handled keeps increasing, the need for more work in big data is also on the rise. The overall objective is to analyze the current optimized solutions / methods/techniques and improve them or invent something new based upon much better performance metrics.

**REFERENCES**


[10] Stepwise Multiple Linear Regression Analysis http://marketing.byu.edu/htmlpages/books/pcmds/REGRESS.html


