ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 9(11), DOI: 10.17485/ijst/2016/v9i11/80464, March 2016* ISSN (Online) : 0974-5645

Comparative Analysis of Information Extraction Techniques for Data Mining

Amit Verma[1*], Iqbaldeep Kaur[1] and Namita Arora[2]

[1]Department of Computer Science and Engineering, CGC Landran, Mohali - 140307, Punjab, India; dramitverma.cu@gmail.com, iqbaldeepkaur.cu@gmail.com
[2]Department of Computer Science and Engineering, Chandigarh University, Mohali - 140413, Punjab, India; arora.ernamita@gmail.com

Abstract

Background/Objectives: This paper emphasizes the evolution of data processing adroitness to advanced data processing taxonomy from Mesolithic to recent years and a comparative study of prevailing tools/techniques which are useful for mainly the analysis of the bulky data. **Methods/Statistical Analysis:** There are various kinds of methods adapted by researchers for analysis of large amount of data. Each method varies on the basis of their different parameters and datasets according to their needs. These methods are implemented on HDFS, Mapreduce and Hadoop environment with integration of R tool. Some Methods are enhanced by the sentimental analysis through NLP which increase the performance of density analysis. **Findings:** The data or associated facts have been in existence right with the birth of human species. It commenced with manual illustration and gradually advanced through current state-of the art storage and processing. Big data involves novel techniques to manage information within limited run time. Big data is acutely beneficial in ventures growth, society incumbency and scientific research. The paper provides an overview of state of the art and focuses on the usage of conventional tools as well as advanced tools and techniques for effective information extraction. **Applications/ Improvements:** To handle this prodigious data, there is a need to upgrade from the traditional data filtering techniques and adopt the new big data diagnostic tools.

Keywords: Big Data, Data Analysis, Data Mining, Evolution, Techniques, Tools

1. Introduction

The usage of data and impact of the internet on business productivity, government services and human lifestyle is inevitable. This internet has a new transformed trend called Big Data, for analyzing the bulky data of repositories, data warehouses used with D_M tools and technologies. D_M is an amalgamation of database and contrived tiding technologies which is an extraction of hidden predictive information from databases. According to McKinsey's report[1], the vigour care, thriftiness controlled, narrates production and private site data fields are revolutionized inheriment of big data[34]. Data is measured, collected, reported, analyzed/visualized using graphs and images. Datum is transformed into an appropriate clump for rectification and hunk of information generally aligned in a significant way. Concerning today's data processors and conveyance way, facts and figures are reconstructed into binary to digital anatomy. It may occur in a diversity of arrangements as bits and bytes accumulate in digital memory, as digits, characters on scrap of paper[2], or as facts are cumulate in creature's mind.

Data is aggregation of files which are massive in numbers/composite in nature; it comprises both organized and unorganized form of data. It comes from worldwide technology like sensors[3] for grabbing the information of weather, putting up some comments, status on sites of social media, videos and depicts etc. Data is elucidated

* *Author for correspondence*

as characterization of facts, notion/specifications in a validate approach acceptable for conveyance, exposition, transformed by personage and computerized[3]. Word data has been acquired from the dual format of ancient Greek term 'Datum' which is expedient as 'to give'. It arises at mid-17th era. Data is contributed as unprocessed factuality. The processing methods are shown in Table 1. It represents the year wise approaches used in different time spans, the phraseology of method and technique which support to handle the bulky data very easily.

Table 1. Data processing methods

Era	Terminology		Technique	
	1.1	1.2	a	b
1800-1890	M_D RK_{CV}	SG_{SY} G_S/W_F	M_U	C_V
1891-1950	M_{CH} CA_{LC} A_{BS}	S_{FP} T_{WD}	M_{CH}	DG_{PrG}
1951-1970	E_{CT}	M_{TH}	BH_{PG}	STA_{PG}
1971-1980	DG_{PG}	M_{SK}	RT_{PG}	MT_{UR}
1981-2004	AT_{OD}	D_M	NU_{NT}	M_L
2005-2015	BG_D	C_{CP}	DB_{CP}	PL_{CP} GD_{CP}

M_D- Manual-Data, SG_{SY}-sign-symbol, RK_{CV}- rock-carving, G_S/W_F-Gestures-written form, M_{CH}-Mechanical, S_{FP}-Solid-Form-paper, T_{WD}- Typewritten-data, A_{BS}-Abacus, CA_{LC}-calculator, E_{CT}-Electronic, BH_{PG}-Batch-Processing, STA_{PG}-Statistical-Processing, DG_{PG}-Digital-Processing, M_{SK}-Multitasking,RT_{PG}-Real-Time-Processing,AT_{OD}-Automated, D_M-Data-Mining,NU_{NT}-Neural-Network, M_L-Machine-Learning, BG_D-Big-data, DB_{CP}-Distributed-Computing, M_U Manual, C_V-Carving, DG_{PrG} Digital Pressing, M_{TH}-Mathematical, MT_{UR}-Multi User, C_{CP}-Cloud Computing, PL_{CP}-Parallel Computing, GD_{CP}-Grid Computing.

Data is measured in different values and weightage like K_B, M_B, G_B, T_B, P_B, E_B, Z_B and Y_B. Each field has different types of data i.e., in medical science; the data related to biological scans, reports for arteries, diseases, medicines, blood cells. In computer science, the data is related to Binary values, Digital data, interconnection data, networking data.

Earlier data was a special and rarefied term. In ancient time no one had even heard the word data and all the people used to rote memorize their thoughts, personal information, business dealing information, king's commoner's information using their biological capabilities. Then, it took the form where they would be writing and managing their data on stones, leaves and maintained cupake. The Evolution of data is shown in Table 2. This table describes in different range of years and each range comprises the phrasings and different techniques. These techniques are further derived in different types.

Table 2. Data evolution (A)

Sl. No	Era	Terminology	Techniques
1.	2500BC-1300BC	CV_{ART}	C_V E_{BG}
2.	1400BC-500BC	TY_{SK}	MC_{CD}
3.	400BC-185BC	PY_{US}	STA_{PG}
4.	186BC-1279AD	P_R	W_T S_{CH}
5.	750-1526	PT_{PS}	M_L

The range from year (1500-2007) has been shown in Table 3, in which techniques and terminologies have been refined to accumulate and manage the data in a better way.

Table 3. Data evolution (B)

Sl. No	Era	Terminology	Techniques
1.	1527-1857	PH_{CRD}	P_G
2.	1856-1960	C_{PT}	BY_{CD}
3.	1961-2001	ST_{DV}	ASCII EDBCD-II
4.	1960-1990	I_{NT}	SRC_{TQ}
5.	1987-2007	MU_{AG}	A/V_{SrD} U_P/D_G

Table 4. Data evolution (C)

SL. No.	Era	Terminology	Techniques	
			1	2
1.	1990-2000	D_M	A_R C_{LS}	C_{LG} R_{EG}
2.	2001-2004	BG_D	NU_{NT} SO_{NA} D_M O_{PZ}	NLP G_A $LG_{SP/UP}$ M_L
3.	2004-2015	SO_{AG}	VZ_{AP}	NN_{AP}

CV_{ART}-Cave-Art, TY_{SK}-Tally-Stick, MC_{CD}-Mnemonic-Code, PY_{US}-Papyrus, P_R-Paper, PT_{PS}-Printing-Process, PH_{CRD}-Punch-Card, M_{CH}-Machine, C_{PT}-Computer, BY_{CD}-Binary-Code, ST_{DV}-Storage-Devices, I_{NT}-Internet, MU_{AG}-Multimedia-Age, D_M-Data-Mining, BG_D-Big Data, SO_{AG}-Social-Age, SRC_{TQ}-Search-Technique, O_{PZ}-Optimization, SO_{NA}-Social-Network-Analysis, $LG_{SP/USP}$-Supervised/Unsupervised Learning, VZ_{AP}-Visualization-Approach, NN_{AP}-Neural-Network-Approach, E_{BG}-Embossing, W_T-Writing, S_{CH}-Scratching, P_G-Punching, A/V_{SrD}-Audio/Video Streaming data, U_P/D_G-Uploading/Downloading, $LG_{SP/UP}$-Learning Supervised/ Unsupervised.

The range from year (1990–2015) shown in Table 4, in which the new emergence trends and technologies were discovered for growth of research technologies and business production. Typically in ancient times, there were two types of material, soft and hard for data to be written.

Pebbles, alloy, shells and stoneware were the illustrations of hard material. People used to write and record the information using engraving, embossing, painting and scratching materials. Wooden board, palm leaves, leather, cotton clothes, birch bark were the instances of soft materials. Each methods and approaches are categorizes in different time periods. With the passage of time new techniques came into being for data handling.

1.1 Mesolithic Period of Data

In this era, Humans used to paint their data on cave walls. Cave art may have begun from around 30,000 years ago. The paintings depicted the living style of creatures who resided in the caves as well as in the surrounding. This also includes religious and social messages as can be seen on a historical monument like Taj Mahal. This age-old tradition of keeping records took the form of presentation and storage of data formally. Information used to be in terms of a variety of contexts, viz., time, quantity, count, days of week, geographical distance, specific places which took the form of number system, time system, unit's standard established later on.

1.2 Vedic Period of Data

During 500BC the tally stick used to be a mnemonic device to platter and report digits, amount or identically memorandums. Tally sticks firstly appeared as beast bones carven with snick in the upper Pal Eolithic which evolved into recording data on piece of wood or single TY_{SK} of wood. TY_{SK} also proved to be effective means for communication of data along with the authentication as the data was recorded on lengthy sticks[2]. These sticks were broken into two halves with same notches (also called it as split tally) each of which was given for identity proof to the parties involved in the communication. With time this approach was dispersed in several methods. After some refinement, there was a way for writing the data i.e., Papyrus. It indicates a dense paper-like stuff made from the core of the papyrus flora. Papyrus is the first known to have been used in ancient Egypt. This papyrus paved way for the invention of what we call 'paper' today.

1.3 Kushan Empire (185BC-'13) of Data

In 100BC paper was invented and word "paper" was derived from Latin word papyrus. This improved the process and material for data writing and tread path for new means to store and manage the data. The data got better readability since the paper now became a thin/flat material produced by the compression of fiber. Paper today is being acquired with an extensive diversity of attributes, reckoning on its signified consumptions for data.

- For constituting valuates like paper currency, greenbacks, bank check, certificate papers and coupon/slates.
- For stow Data/information like tome, exercise book, journals, newsprint, artistry, hardback, archives.
- For exclusive use like logbooks, notes for memorizing and short-term use like scrape papers.
- For wrapping like rumpled box, sack, envelops, twining paper and wallpapers.
- For circulation between someone and enclaves.

1.4 Medieval Year of Data

After the establishment of paper as a medium to write upon, the printing process came into being in 1440. Printing is the process of reproducing text data, images, and patterns using a master form or template. Evolving of the printing the data into the form of hard copies was generated and that was the stage where the data started increasing rapidly. So that has become a big challenge for storing or placing the data properly. To overcome this problem researchers have to think about digital conversion and usage of data. In 1750s, Punch card was discovered.

Punch card is a wedge of rigid paper that obtained either instructions for handling mechanized or data for refining information. Both instructions and data were considered by the existence/non-existence of holes in already clarified region. The algorithm of punch card provides its procedure and described as the successive meantime for extensive weighted least square solution[5]. It was used for grant corporations to keep and retrieve information by enrolling it into the computer. Afterward this approach was restrained in different ways. It was the elementary way of keeping and accessing data in the ancient 1900s, and its initiation was then replaced by other approach in 1960s which has now become a rarely known thing.

Algorithm Punch card

Begin Initialize $s_{n\,x\,k}$ // successive intervals analysis is an n x k

For(T_{able}=1,T_{able}≤$T_{able(total)}$,T_{able}++) // T_{able} table number of times and each table T_{able}€ s_n

{

Stuffed F_{ib} //where b is the category boundary forith stimulus

}

For (R=1, R ≤ R_n, R ++)

{

For (C=1, C ≤ C_n, C ++)

{

Punch C_{n-k} // where k>1 and k<n

}

Generate cards G_c //where G_c < n x k, R €n and c€k

} // n individuals ofR*C (row and column)

If

{

M_{card} contain F_{ig} //where F_{ig} € Z_{ig} Since the data are rarely available

Display "Punched Successful"

}

Else

Exit(1)

Apply sorting according to G_c

If

{

$G_{c(n)}$ ≤T

remove all other punches

}

Else

perform operation

apply desk calculator

end

After that in the world of digital data, the invention of computer proved to be the turning point of the modern technology from 1820. In late 1980's people started depending on computers to handle the information and possess the capability to store large data in memory. Along with the development of computer evolved/generated many new storage devices with advanced data processing and repositories. Table 5, shows the different mechanisms established in different years so far.

1.5 Web based Year (1990-2015) of Data

The Internet era began in mid-1990s. Internet has been a pivot of success of the people. It provides the amazing facility of searching any information from any nook and corner of the world by just at the click of a button such as communicating with E-Mail, surfing search engines, using social media websites, accessing web portals, opening informative websites. In 1990's the data mining method was discovered for refining the data into knowledge and categories in M_L, A_{RI}, and STA_{CL}. After filtering the data researchers faced many problems regarding accumulators/repositories essentially which are available as ubiquitous manner. So to overcome this challenge, in 1999 cloud computing concept came into existence and it was pioneered by Salesforce.com[6]. It primarily provides the information resources as a service and can be acquired only with internet usage.

The internet has changed entirety of the corporation services, working manner, government justifications, and humans' way to live. But many of the novel approaches are metamorphosed as "big data"[35,36]. This revolutionized method aroused a lot of information with some conflictions and placed around ubiquitously. Only a quarter of the total available information has been put into digital form till the year 2000.

According to Gartner's Report, there was a rapid growth of digital data doubling around every two years. The report further says that if all the digital information

Table 5. Device discoveries

Years	DS_{CV}	Year	DS_{CV}	Year	DS_{CV}	Year	DS_{CV}
1877	P_{GH}	1956	HD_{DK}	1971	FY_{DK}	1997	MA_{CD}
1898	T_{GH}	1963	MC_{TP}	1980	CD	2001	USB,SD_{CD}
1928	MG_{TP}	1968	TW_{MY}	1984	CD-ROM		
1932	MG_{DM}	1970	BU_{MY}	1995	DVD		

DS_{CV}-Discoveries,P_{GH}-phonograph,T_{GH}-Telegraphone,MG_{TP}-Magnetic-Tape,MG_{DM}-Magnetic Drum,HD_{DK}-Hard-disk,MC_{TP}-Music-Tape,TW_{MY}-Twistor memory, BU_{MY}-Bubble-memory, FY_{DK}-Floppy-disk, MA_{CD}-Multimedia-card,SD_{CD}- SDCard

were kept or stored in the compact disks and placed in the form of stack with six different knots then these flocks would extend to the Moon. Such is the volume of this massive digital data.

2. Existing Tools and Techniques

A spacious heterogeneity of approaches and automations have been originated and altered to capture, curate, examine and envision of big data. The skills depict from innumerable domains like computer-aided, applied mathematics, statistics, transportation logistics and economics. A few proficiencies and high techs were originated in digital globe to handle/address limited variety and volume of data but the growth of organizations were continuously adapted by novel techniques. There is a group of tools and techniques that have been developed by research community and data analyst[7]. Tools provide set of methods and algorithms that help in better utilization of data available to researchers and inherit data quality from past to present. Comparative study of tools is carried out by some open source tools freely available on the internet.

2.1 Data Analytical Tools

The following is an explanation of the advanced tools in data mining. These tools are useful analyzing data and extracting information. Orange and Rapid Miner are the tools mainly used for Knowledge extraction. Orange is a constitutive based data mining, examine and M_L tool. It is written in python. It consists of a canvas interface onto which the user places widgets and creates a data analysis workflow. It is beneficial for text mining and bio-informatics.

Figure 1. Data-End point level-0.

Where-as R_M allows a GUI to design/accomplish analytical productivity. It is written in java. It provides an integrated environment for M_L, D_M, T_M, P_{RA}, STA_{AY} and entire abuses of the D_M process such-as visualization, validation and optimization. Client server model is used in R_M tool with offering as cloud services. Figure 1, gives simple representation of data processing method with help of D_M tools, techniques and concludes that techniques are to consider as function.

KNIME is Konstanz Information Miner and an affordable tool for data examine, covering and consolidated components of M_L and D_M through its standard pipelining approach. It represents a GUI and accord gathering of knobs for data pre-processing. By using this tool users can optimizing the creation of data flow and execute selective paces. It is written in java, based on Eclipse and not a scripting language. KEEL is Knowledge Extraction Evolutionary Learning and a clump of M_L software tools. Spanish National Project originates the KEEL tool and a module of java programming which is used for heterogeneity KDD tasks. It provides a GUI for user to interacting with dissimilar datasets

R, Weka and Rattle-GUI are tools used for Knowledge abstraction, where R is an environment and a Language for STA_{AY} and visualizations. Environment operates with OS like UNIX, Windows/Mac. R provides functions through handling; accumulate bulky data or computing calculations[8] on matrices effectual programming language followed as conditional algorithmic functions whereas weka is a clump of M_L apps which are written in the java. Waikato Environment is for information examines/analysis of M_L algorithms for D_M. Explorer is the primarily interface of the weka. Its add-one functions are pre-processing paces, M_L algorithms, assess methods. It can support some import files such as CSV, ARFF, C4.5 and Binary. And Rattle-GUI is easily accessible software which allows a GUI for D_M using the R STA_{PROG}[8]. Through GUI it provides D_M functions and to divulge the ability of R_{STA} tool.

NLTK, Apache Mahout and SCaViS/DataMelt tools are used for batch processing in which, NLTK: The Natural Language toolkit is the combination of distinct libraries and number of programs that which supported as parabolic and STA_{NLP}, Its platform is written in python. NLTK deals with the data of human language. Apache Mahout name comes from its close association with apache Hadoop which uses an elephant as its logo[9]. In Figure 2, gives a simplest/easiest way of describing process in details. In which all techniques are mined the data in appropriate manner and further categorized in different types.

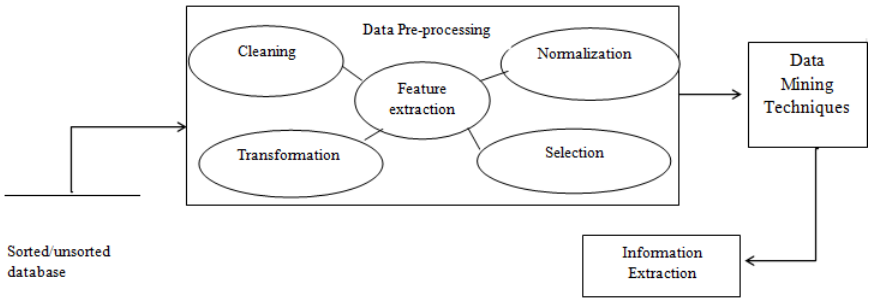

Figure 2. Technique level-1.

Hadoop is an easily accessible framework from Apache which give authorization to accumulate and deal with big data[37,38] in a distributed system by using basic programming models[27]. It is primarily used for creating scalable M_L algorithms. It implements popular M_L techniques such as Recommendation, Classification and Clustering. It started-as a sub-project of Apache's Lucene in 2008. Mahout became a top level project of Apache in 2010. Twitter uses Mahout for user interest modeling[22]. Yahoo! uses Mahout for pattern mining.

SCa ViS/Data Melt is a collaborative framework which is applicable for data analysis, visualization and a platform for performing various computations. It is written in java and streams up on various types of OS in which JVM must be installed. It supports an interactive design which plots in 2D and 3D views. DataMelt tool is assist with high level programming language in which are accompanying as Groovy, Java, JRuby and Jython. The name of DataMelt was renewed as SCaViS.

Mlpy is machine learning python is written in python language and support M_L libraries. It provides the solution of supervised/unsupervised troubleness with the help of M_L methods. It categorized in different versions and latest version is 3. 5. 0. It primarily used in bioinformatics field.

In Figure 3, the rigorous flow of data in database through various D_M operations and these are categorized in different techniques such-as association rule, classification and other techniques are producing-output in form of statistics which are converted to graphical representation.

Fityk is an application for data analysis, adjustments of arcs and support normal distribution, Gaussian and sigmoid functions which are provided-as bell shaped functions and required for plotting an arc/curve. It is written in c++ language, use wx-Widgets. It affirms with the python and scripting languages. GATE is General Architecture for Text Engineering tool was earlier acquiring at the Sheffield University in 1995. It generally used for academic and research purpose. It binds up with NLP and information extraction. It is written in java and supports other type of languages. It assumes import files in several formats which are included-as txt, pdf, xml, html and java consecutives.

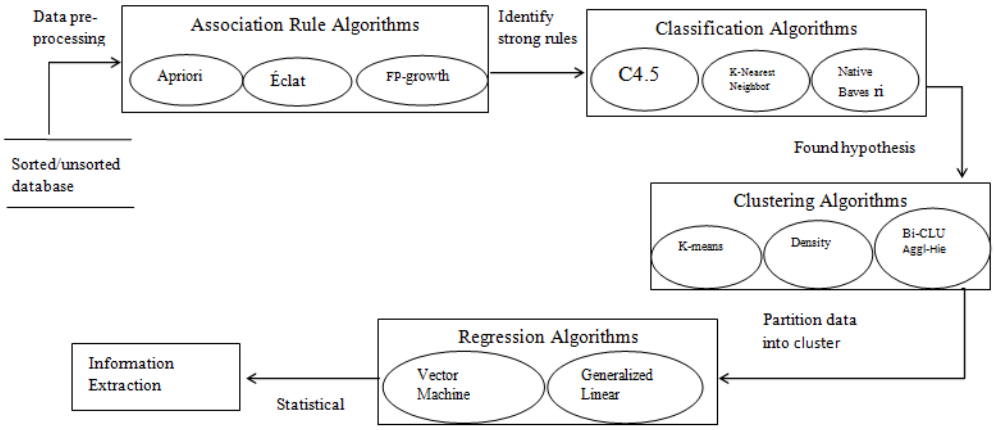

Figure 2. Technique level-1.

2.2 Data Analytical Techniques

Techniques are primarily use for mining information from enormous data. Contemporary issues and arising computing potentiality will inspire the development of novel techniques which are easily extracting the information in limited time period and developed for solving the problem of big data[39,40].

Association Rule is a set of techniques for attaining interesting relationships between parameters in databases. It consist different possible test rules and develop different kinds of algorithms. A_R technique is used in market basket analysis, in which retailer can determine the product which are frequently bought together[11]. They are categorized in such-as multilevel A_R, multidimensional A_R and quantitive rule.

Classification is determining the categories of class labels and a function for generalizing the structure of contemporary data. For Instance classify the e-mails as-spam if it might be not relevant. In Prediction model used to predict the unknown and missing values, where it used in persistent evaluate function. It categorized in the form of supervised/unsupervised learning. In supervised the set of training data consisting as vectors and supervisory signals as desired output[12]. In unsupervised the hidden structure finds in unlabeled data and in statistics there is a problem of frequency estimation. It classified-as decision tree induction, Support-Vector Method, Bayesian-Classification, Neural-Network. Clustering[28] is used to make different clusters on the basis of certain parameters. It splits up the objects depending upon their shapes, features, categories and behaviour. Clusters must have some similarity which is evaluated using specific metrics[1]. It is a statistical method to analyse the objects and categorized into various methods such-as partitioning, model-based, hierarchical agglomerative, density based and grid-based.

To extracting information by the help of distinct number of techniques are shown in Table 6, comprises the techniques, algorithms, sub-type, specific application, pro/cons and arranging in the manner of popularity.

Regression is used to transform the value of dependent variable into independent variables. It is used for prediction such-as to predict weather forecast on the basis of these parameters wind, temperature, clouds, tornado, lightning etc. Its task begins with identification of some target values in the data sets and categorized as multivariate nonlinear and linear multivariate Regression. More details of Table 6 found in Appendix. In order to implement these techniques/tools which have been described in detail in the following section.

Table 6. Information extraction techniques

Sl. No.	Name	Algorithm	Sub types	Specific app	Pro	Cons
1	A_{RL}	AP_{AO},EC_{AO} FPG_{AO}	MRA_R,CBA_R BA_R,QA_{Rc} FZA_R	Na	E_{IM},LI_{PY}, E_{PRL}	SAP_{AO},TR_{DB} SC_{DB}
2	CU_{LG}	K_M,D_B,BC_{LG}, AG_{HR}	PA_{MD},HR_{MD}, DB_{M-D},GB_{MD}, MB_{MD},CT_B	AG_{NES}, DI_{ANA}	CL_{AG},FK_M $SCK-M$,CSC_{DS} $RY_{SC/SI}$	$HS_{PH/NO/OU}$$FE_{LO}$,$NC_{VS/DY}$ ODK_M,$H_{NOD/OU}$
3	C_{LS}	C4.5,NK_{AO}, NB_Y,DC_{TE}	SP_{BA},ML_{BA}	OU_{ANA}, EV_{ANA}, G_A, R_{SA}, FZ_{SA}	$DT_{UD/IP}$, $H_{NM/CD}$, ATC_{LS},R_{CX}, SLR_Z	DT_{NV},$SKR_{XV/SW}$ LR_{BM},UNB_Y
4	RG_{EG}	GL_{MD},V_M,LR_{EG}, PR_{EG}, LGR_{EG}	Na	Na	$PD_{SH/LG}$, WA_{PP}	$PZ_{DF/HD}$,D_{EX}

A_R-Association, A_{RL}-Association-rule, AP_{AO}-Apriori-Algo, EC_{AO}-Éclat(Equivalence-class-transformation)Algo, FPG_{AO}–Frequent-Pattern-growth Algo, MRA_R-Multi-Relation AR, CBA_R-Context-Based-AR, BA_R-Binary-AR, QA_R-Quantitative-AR, FZA_R-Fuzzy-AR, DI_{ANA}-Divisive Analysis, CU_{LG}-Clustering, K_M-K-means, D_B-Density-Based, BC_{LG}-Bi-Clustering, AG_{HR}-Agglomerative-Hierarchic, MD-Methods PA_{MD}-Partitioning, HR_{MD}-Hierarchical, DB_{MD}-Density-based methods, GB_{MD}-Grid-based, MB_{MD}-Mode-based, CT_B-Constraint-based, AG_{NES}-agglomerative nesting.

The first D_M tools found on records have been introduced in 1990s. Initially, D_M concepts were used to handle very small data and that data was generally structured. In Table 7, the D_M tools in the range of year 1900 to 2000 have been enlisted and represent the language, features, establishment-date, salient-features and limitations of the different available D_M tools.

More details of Table 7-9 found in Appendix. With an evident rapid growth of data, the existence of unstructured data in the form of text, audios and videos has been posing even greater difficulty to extract relevant data. In Table 8 shown the techniques which are developed during the years (2000 to 2005) with their factors. The challenges of data are being raised at even harder levels with the structured, unstructured, semi-structured and real-streaming data which is growing every second.

The Table 9 represents the range of data mining tools in 2005 to 2015, which have inherited features of conventional tools and have also added up new robust features for extracting information in fast and effective manner.

The above sections have described the tools and techniques popularly being used in the D_M process. The area of data extraction has attracted many researchers to exploit their creativity to mend the process to become more effective, efficient and faster. One of the current research work by[13] reviewed the prevailing results of compact size files. In this paper the small files of HDFS

Table 7. Data mining tools (1900-2000)

Tool	Release date	Written in	Type	OS	License	Features	Pros	Cons
W_K	1993	J_V	M_L	Lx, OS-,Ws	GNU_{GPL}	CLS/REG_{ALGO},CLU_{ALG} $A_{R-LALGO}$,D_{PT} $GUI_{EXDA/EXP/KF}$	ES_U RM_{EX}	OZP_{PR}, DOC_{PR} STA_{WK}, CR_{WK}
G_T	1995	J_V	T_M, IN_{EX}	C_{PF}	L_{GPL}	$BS_{LR/PR}$,C_{LIB}, FK_{GDE},OC_{DDS} AN_{NIE},GUI_{JB}	LE_D, E_{BDC}, $EXO_{MAT/DBT}$, Q_E, R_Y	DB_{SW}, ARC_{DEG}
R	1997	C,F_N& R	STA_{CP}	C_{PF}	GNU_{GPL}	D_{EX},OT_{DE},C_{LT},TX_M TS_{AY},SN_{AY},PL_{CP} GP_{VZ}, BG_{Wapps} D_H&E_H,A_{LA}	PY_{STA} R_O,R_E	$L_{DM/ALAK}$
RT_{GUI}	1997	R_{PRG}	GUI_{TK}	Lx, MAC,Ws	GNU_{GPL}	$IFC_{SV/EXC/R}$,$STA_{MIN/MAX}$ $MDL_{DT/RF/LR}$,$EVA_{CM/RC/CC}$ $CHT_{BP/HG/CR/DG}$,T_{RF}	PIC_{RP} MA_{GUI}	PIC_L

J_V-Java, W_K-Weka, F_N-Fortran, STA_{CP}-Statistical-Computing, C_{PF}-Cross-Platform, GNU_{GPL}–General-Public-License, M_L-Machine Learning, RT_{GUI}-Rattle GUI,$_{PRG}$ -Programming, $_{TK}$-Toolkit, $G_{T\text{-}GATE}$, T_{M}- Text-mining, IN_{EX}-Information-Extraction, Ws-Windows, S_Y-Scalability, DB_{SW}-Database-Slow,CLS/REG_{ALGO}-support-classification/regression-algorithms, D_{PT}-data-pre-processing-tools, CLU_{ALGO}- clustering-algorithms, A_{RLALGO}-Association-rules, D_H&E_H- Data & Error Handling.

Table 8. Data mining tools (2000-2005)

Tool	Release date	Written in	Type	OS	License	Features	Pros	Cons
N_{LTK}	2001	P_Y	NLP	Lx, MAC, Ws, OSx	ACH2.0	M_{TR},G_{TR} CLS_{TX-},SM_{FLT}	H_{RDLY},OOP_{ES}, ES_{EX},UC_{SG}, LIB_{POW}	CT_{WD}, TKZ_{TX}
K_E	2004	J_V	E_R,B_{IN},D_M	Lx, OS, Ws	GNU_{GPL}	S_Y,I_{UIF},H_{EX},B_{EX} $WF_{IMP/EXP}$,P_{EX}, API_{PL}	MO_{AY}, MS_{SPY} CY_{DVK}	EMT_L, OZP_{PR}
K_L	2004	J_V	M_L	C_{PF}	GNU_{GPLV3}	CLS_{DV},CLU_{DV}, D_{Vz},REG_{DV},A_{RDV}, DV_{Vz}, FLY_U, LRN_{EVO}	EV_{ALGO} Fz_{ST}	L_{ALGO}
FIT_{YK}	2004	C++	C_{FG}	C_{PF}	GNU_{GPLv2}	DS_H,EQ_{CT} D_{MU},AT_{SCP}	$WLSQ_{LAGM/NM/GA}$ $GP_{PLS/PRS}$	DS_{MPH}
S_{CV}/D_{TM}	2005	J_V, P_Y	D_{AY}	C_{PF}	GNU_{GPL}	STA_{CAL}, $D_{2D/3D}$, $F_{HT/CH/AY/CAY/NN/FZ}$	VZ_{EXP}, $B_{NS/MAY}$, $W_{STA/ARI}$, $F_{MTH/NCAL/VZ}$	LKJ_Y

N_{LTK}–Natural-language-toolkit, P_Y-Python, NLP-Natural-Language-Processing, K_E-KNIME, J_V-Java, E_R-Enterprise-Reporting, , B_{IN}-Business-Intelligence, D_M-data-mining, GNU_{GPL}–General-Public-License, K_L-KEEL, M_L-Machine-Learning, CPF-Cross-Platform, S_{CV}/D_{TM}-SCaViS/DataMelt, CFG-Curve-fitting, Ws-Windows, M_{TR}-Machine-Translation, G_{TR}-Google-Translate, CLS_{TX}-Text Classification, SM_{FLT}-Spam-filters, S_Y-Scalability, I_{UIF}- Intuitive-user-interface, H_{EX}-High-extensibility, A_{RDV}-Association Discovery.

Table 9. Data mining tools (2005-2015)

Tool	Release date	Written in	Type	OS	License	Features	Merits	Demerits
R_M	2006	I_{LA}	STA_{AY},D_M, PR_A	C_{PF}	$AGPL_{PY}$	D_H/A_P, MA_{CV}, GUI_{IV}, F_{OD}	$S_{VZ/STA/AS/OD/PO}$	DB_H
O_R	2009	P_Y, C++, c	M_L,D_M DVz	C_{PF}	GNU_{GPL}	V_{PG},V_Z,I_{TC}&$D_{AT}L_{TX}$, S_{ITP},EX_{DOC}	B_{DEB},SH_{SC} S_{NVE}	BG_{INS}, RP_{cpL}
MA_{CH}	2011	J_V,S_{CL}	M_L	C_{PF}	ACH2.0	ENV_{DS},DM_{BG}, BG_{AYQ},$IC_{MX/VT}$,FIF_{DS}	S_Y,NO_{ALGO},U_{IM-},M_{PTR} R_{FE}&H_{BE}	OH_{BE}
ML_{PY}	2011	P_Y,C++, C,	M_L	Lx, MAC, Ws, OSx	GPL	$CLS_{LDA/LR}$ $REG_{LSQ/RR}$ $CLU_{HC/KM}$ DMR_{EDAY}	$SP_{VM/PLS/MHC}$	PA_{LM}

O_R-Orange,M_L-Machine-Learning,D_{Vz}-Data-Visualization,C_{PF-}Cross-Platform, GNU_{GPL}-General-Public-License, R_M-Rapid-miner, I_{LA}-Language-Independent, STA_{AY-}Statistical-analysis, PR_A-Predictive-analytics, $AGPL_{PY}$-Proprietary, J_V-Java, Lx-Linux, Ws-Windows, F_N-Fortran,W_K-Weka, A_{CH}-Apache, MA_{CH}-Mahout,D_{AY-}Data-Analysis, S_{CL-}Scala, S_{Y-}Scalability,C_{PF-}Cross-Platform,V_{PG}-Visual Programming, V_Z-Visualization, D_{AT}-Data-Analytics, I_{TC}-Interaction, L_{TX}-Large-toolbox, S_{ITF}-Scripting-interface.

which are accumulates in storage and analyses for finding the flaws, volume distributions of small files are based on its consistency. The author proposes the small file merging algorithm based on balance of data block and gives integrated algorithm. Finally, the paper concludes with experimental analysis on the proposed algorithm and proves that algorithm by reduces the memory consumption at the crucial nodes of collections and improves the efficiency of data processing of collections.

Algorithm TM (Tetris Merge)

Initialize F_{set}, q_{fl} , q_{tl}
For (F_{set}=1, F_{set} <M_{limit}, F_{set}++)
{
// Merge of the files where merge limit is the maximum limit
//F_{set} € *Smin (minimum free space)*
if
{
f!=0;
cout<<"select maximum q_{fl} , q_{tl} ";
Select a queue *qt* from *qtl* , push *f* into *qt* , push *qt* into *qfl*
}
then
f ==*qmax* // *qmax* € *qfl*
// Push f into *qmax otherwise*
exit (1)
else
{
f==NULL
Perform FM_{limit} // FM_{limit} Merge remain files to ***MergedFile*** in queues
}
} exit

TM algorithm has been compared with other two algorithms of file import using the comparison of time consumption of importing files into HDFS. Comparison shows that memory consumption at name node is reduced by 31.4%, as compared to other algorithms. On the whole, the author reduces the speed, memory consumption and time consumption as compared to other import file algorithms

The implementation of Tetris Merge algorithm gives experimental result on the basis of three parameters such as by importing small files represents in Table 10, by processing speed in Table 11 and by memory consumption in Table 12.

Table 10. Comparison of importing small-files

ALGO	VOF	File After Merge	Time CS_{PT} (s)
N_{IM}	4294/10.12 GB	4294	567
MFM_A	4249/10.12 GB	82	249
TM_A	5249/10.12 GB	84	252

N_{IM}-Normal-Import, MFM_A-Monofile-merging-algo,TM_A-Tetris-Merge-algorithm, VOF-Volume-of-file, M_{SG-}Map-Stage, RD_{SG-}Reduce-Stage, CS_{PT}-Consumption

Table 11. Comparison of processing speed

ALGO	Time CS_{PT} of M_{SG} (s)	Time CS_{PT} of RD_{SG} (s)	Total time CS_{PT}
N_{IM}	13,993	333	14,326
MFM_A	1190	119	1309
TM_A	1035	101	1136

N_{IM}-Normal-Import, MFM_A-Monofile-merging-algo,TM_A-Tetris-Merge-algorithm, VOF-Volume-of-file, M_{SG-}Map-Stage, RD_{SG-}Reduce-Stage, CS_{PT}-Consumption

Table 12. Memory consumption of name node

ALGO	Data Blocks after merging	Memory CS_{PT} at Name node (byte)
N_{IM}	4294	644,100
MFM_A	245	36,750
TM_A	168	25,200

In[14] proposed the usage of context awareness scheduling by improvement of apache Hadoop and provides comparative performances on the basis of time parameter and standard deviations, the experimental results shows the positive impact of performance. In Table 13, shows the analyses of elements which are affected by the context-aware scheduling with four cases.

Table 13. Context-aware scheduling

Case	Total-Map-Time (s)	Average-Map-Time	Map-tasks-Standard Deviation	Speculative-tasks
A	149	39.47	15.73%	2
B	788	222.97	59.86%	1
C	348	38.38	18.09%	3
D	477	68.42	29.91%	1

A-Simulates a dedicated Hadoop-cluster, B-nodes used for other purposes, C-nodes are shared with other applications, D-provide extension of applications.

In[15] describe the methods and surroundings for analytics on cloud for big data analysis and identify the gaps between technologies. The paper gives complete scenario on three key groups such-as data management, developing model, validation and user-interaction. The usage of cloud resources being enables and strengthens big-data area.

In[16] present the complete survey on yesterday, today and tomorrow of big-data. Authors mention the troubles and overall assessment of big-data. The paper gives the introduction on big data features, classification, techniques, process and applications. The author also increased the reliability of accessing and handling the big data.

In[17] proposed the method for processing data in parallel manner as small-clusters in distributed-chunks and enhanced by the sentiment-analysis through NLP. In this paper the proposed model increased the performance of complexity analysis. The author also evaluates the experimental result by comparison of existing system, proposed system and transformed unstructured-data into structured-form by using proposed method on Mapreduce technique.

In[10] focused for analysis of collected tweets on Hadoop based framework and ecosystem and transformed the result into graphical-charts as representing purpose. Hadoop-ecosystem is core functionality of Hadoop. The author represents the comparison between the HBase-component and Traditional-RdBMS.

In[18] has worked for mapreduce, its advantage, disadvantages and integration with other technologies. The paper clarifies the improvement techniques for mapreduce which can handle big data in less run time and described the complete series of steps for mapreduce in which data management and data processing techniques are easily applied.

In[8] proposed the integrated approach for Hadoop and R-tool which are used for processing data. The paper provides three ways to integrate R and Hadoop for big-data analysis with R Hadoop, Rhive and R streaming. Then the author has been integrated to traditional databases like RJBDC, RODBC and Apache Mahout.

In[7] studied as the rise of big data in cloud computing and focus on relationships between big-data and cloud-computing. The paper presents the comparison between storage system of big data, NoSQL databases, several big-data cloud-platforms, big-data categories, its case studies and vendors. The author proposed a visionary view for big-data with classification method and focused on the batch-processing tools and its challenges.

In[3] represents a brief review on big-data problems, opportunities, challenges, current techniques, technologies and proposed various potential techniques to overcome the problem, including cloud-computing, quantum-computing and biological-computing. These technologies are still under development and new emergence trends such as granular-computing and bio-inspired-computing have been discussed. In this paper the author proposed several techniques to overcome the problems of newly emerged techniques.

In[4] proposed a method to improve traditional information handling/retrieval tools by including elements of distributed-processing. The experimental result presents a viable test of mapreduce-framework for pattern matching and data analysis

In[19] proposed a novel language which is dedicated modeling language to reduce the gap between multiformalism and logical modeling, to evaluate the dissimilar abstraction levels by granting permission to the big-data application designer and system administration. The paper represents the complex metaphor into the big-data applications and originates with the SIMTHESys-framework which provides a tool for rapid origination for new formalisms. The author uses the HQLqueries for easily revamp with proposed big data formulism and gives complete translation algorithm.

In[20] presented an algorithm for dividing the frequent 1-sequences and prelarge 1-sequences from original database when some sequences are deleted and to handle the sequential-pattern by adopted the concept of prelarge with sequence deletion. Earlier, only Fast_Updated_Sequence_Pattern (FUSP) was mainly used to handle frequent 1-sequence from database. This proposed algorithm used to maintain and update the FUSP tree. In experimental-phrase the paper represents the performance of proposed algorithm based on time and the number of tree-nodes.

Table 14 represents the list of data-sets and Table 15 represents the comparison of performance basis on thresholds and DR (Deletion-Ratio) set to evaluate the proposed algorithm.

Algorithm Fast_updatation_sequence_pattern

Begin
Initialize D, H_t, p_s, F_tr, Su, Sl , c, T.

// F_tr =*FUSP_tree*, H_t= *Htable*, p_s=*Pre_Seqs*

For(F_tr=1, p_s =1,F_tr <T, p_s++, F_tr ++)
{
Update C-T
//Where function $F = \frac{(U_{AT\ s} - L_{AT\ s}) x |D|}{U_{AT\ s}}$
}
If (C_n !=0)
{ //count number of q from T.
Calculate $S_{T(s)}$// 1-sequences from T.
} //Where s € $S^T(s)$
Do
= –
If
{
s € *Htable* then
call Procedure_
}
Else if
{
s € then
call Procedure_SubCases.
}
Else
Rescan all sequence
End

Table 14. Characteristics of used databases

Sl. No.	Database	#D	#I	AvgLen	MaxLen
1.	DB1	59,601	467	2.51	267
2.	DB2	88,162	16,470	10.3	76
3.	DB3	990,002	41,270	8.1	2,498
4.	DB4	100,000	870	10.1	29

#D-Total transactions, #I-Total transaction, AvgLen-Average-length of transaction, MaxLen-Maximal-length of transactions, DB1-BMSWebView, DB2-retail, DB3-Kosarak, D4-T10I4D100K

Table 15. Pre-large1-sequences under threshold

DB1	0.3%	0.35%	0.4%	0.45%	0.5%
(DR: 10%)	2	4	4	2	3
DB2	0.6%	0.8%	1.0%	1.2%	1.4%
(DR: 5%)	37	21	12	4	2
BD4	1.2%	1.4%	1.6%	1.8%	2.0%
(DR: 1%)	11	12	6	7	4

The formula of pre-large concept is calculated as

$$f = \frac{(S_u` - S_1) \times |D|}{S_u}$$

In[21] presents a HACE-theorem which is characterized as heterogeneous-autonomous complex and evolving relationships. The author proposes a big-data processing-model and features of revolution of big data. Big-data processing-models consist three-tier architecture and each tier have different performance which are consisting as in tier-1, it covers the main platform for data-mining as low level techniques, in tier-2, target on high-level-semantics and tier-3 describes an actual challenges of mining-algorithms.

In[22] convergence on emerging trends and it works on distributed data centers for hosting large data repositories. To handle structure, unstructured, semi structure and real-time-data have been analyzed and equate. Brewer's CAP theorem provides the solution for handling big-data analysis for decision making in business, productions as well as Research academy.

In[23] present a framework to disintegrate the big system into four different consecutive modules. These four modules are such-as generating of data, acquisition of data, storage of data and analytics of data. The paper gives the comparison between the traditional-data and big-data and entire review on big-data evolution, paradigms, its architecture, challenges and steps to handling bulky-data, storage infrastructure and taxonomy of big-data analytics. The authors focus on the value chain of big-data, which covers the entire lifecycle of big-data and gives the overall mechanisms and approaches in different big-data phases.

In[12] present comparative analysis on various clustering techniques which are recently used for analysis of big data, and finding the major techniques which gather information in an appropriate manner. The author reviewed the overall trends in between clustering techniques.

In[24] proposed a novel prediction algorithm by combining the rough sets theory with extreme learning machines. According to its experimental results, the performance of proposed algorithm has higher prediction accuracy and better efficiency as compared to Extreme learning machines. ELM methods have several defects such as it based on the minimized empirical risk and it calculates directly least square solution. Through ELM as compared to proposed algorithm the performance of model is substances affected and poor robust. In Table 16, the proposed algorithm performs with online data sets and removes the duplicate attributes.

Table 16. Three datasets

Data sets	Class	Samples	Condition attributes
Iris	3	150	4
Soy bean	19	307	35
Zoo	7	101	16

In Table 17 performance evaluation for increase the accuracy of proposed algorithm and it presented the conventional ELM algorithm, higher training and higher testing accuracy.

Table 17. Training and testing accuracy comparison

Data sets	ELM		Proposed algorithm	
	Training accuracy	Testing accuracy	Training accuracy	Testing accuracy
Iris	0.9200	0.9067	0.9876	0.9876
Soy bean	0.5033	0.4771	0.6950	0.6804
Zoo	0.9000	0.7200	0.9600	0.9600

Algorithm of integration of rough sets and ELM

Initialize R_{SET} // RSET is raw data set
Select $Data_{(Proc)}$ // $Data_{(Proc)}$ is data processing
If $Data_{(Proc)} = Data_{(Pre\text{-}Proc)}$ // $Data_{(Pre\ Proc)}$ is data pre processing
{
Create DDT // Discrete Decision Table for n attributes
For ($DDT_{(n)}$=1, DDT<$DDT_{(n)}$, $DDT_{(n)}$++)
{
RDT : Reduce Decision Table
}
For (RDT=1, RDT< $RDT_{(n)}$,$RDT_{(n)}$++)
{
Create D_S // D_S Data set
}
Perform Normalization for D_S
For each value of D_S under given value of threshold T
(D_S =1, D_S < $D_{S\ AT\ T}$,D_S ++)
{
Create NN // Randomly generated Neuron threshold
Training of each NN
Perform testing
}
Else
Exit

Big data has been in use nowadays for diverse application domains like improvement of life styles, planning and implementation of smart cities and other developmental activities. In[6] present an intellectual perspective on smart cities based on big data processing and propose a cloud based services that can be integrated with big data and develop decision making in smart future cities. Cloud computing provides better probabilities for manage, process and analysis large amount of data that has been generated in cities. In[25] presented Ophidia (a storage model), the effort of a big data analytics research resolves at reinforcing approach, analysis, and mining of n-dimensional array-based scientific data. The author also reviewed the ways in which this project integrated with conventional data ware house, OLAP framework, novel storage model, parallel paradigms and numerical libraries. These all are follow the order from which the big data analytics are faced as challenges in domaine Science.

Some researchers have done work related to Association data mining technique and integrated it with cloud computing and big data[29–33]. In[9] reviewed and revised a parallel Association Rule ($A_{R)}$ mining strategy integrated with cloud framework and developed the improved Apriori algorithm on Hadoop platform with map reduce programming model. The improved algorithm has been performing better with comparatively less cost than that of original apriori algorithm. The experimental result have been evaluated with online data sets and applied on revised apriori algorithm which improve the performance of proposed algorithm and performance totally depends on the number of Hadoop nodes. Table 18 shows the online data sets and execution performance on the basis of time.

Table 18. Execution time for all data sets

Dataset/ threshold	.95	.90	.85	.80	.75
DS-1	50.26	68.529	88.559	133.621	220.631
DS-2	108.18	156.326	220.122	392.927	878.156
DS-3	65.746	71.005	86.53	91.251	111.547
DS-4	84.612	241.172	681.576	1552.194	2934.792

DS-1-T1014D100K, DS-2-chess, DS-3-mash room, DS-4-connect

Revised Apriori algorithm evaluates results using online real-life datasets. Here, 'Chess', 'Mash room' and 'Connect' are the names of the datasets that have been used and another data set 'T1014D100K' has been synthetically generated data set. Table 17 elucidates the execution time of revised apriori algorithm on different data sets. The performance of dissimilar datasets is tested on single node Hadoop system.

Text mining has found a significant place in conventional as well as current data mining techniques. In[26] design the text retrieval system which is based on dynamic knowledge and improve the question answering information retrieval speed. The accuracy of retrieving the text and the speed have been improved on the basis of some instances that are with catch technology, without catch technology, distributed computing and multithreading. The retrieval time is an evaluative parameter for comparison with various optimization techniques. Table 19 shows the comparison of retrieval time on the basis of different instances which considerably enhance the speed of information retrieval system in less number of time units.

Table 19. Comparison on retrieval time

Instance	No cache	Use cache	Distributing	Multithreading index at same time
Times	About 10s	About 100ms	About 60ms	About 20ms

In[11] presented an approach of optimization technique named as adaptive interval configuration and developed to upgrade the dynamic approach. The candidate pruning technique has been developed using Adaptive Interval Configurations algorithm for mining better association rules. The results have been evaluated by comparing these algorithms, where AIC gives encouraging results than that of Dynamic item set counting and apriori algorithm.

One of the earlier research works by[2] includes an approach for processing data with tabulation system. This formed the basis of effective processing and management of data. The challenge of fast rising population cropped up which led to resolutions and innovations in the whole process of data management using mechanisms such as tabulating system, associated devices like the press, sections of pin and press, the counters and the combination counting for handling data. The storage of data was a dream to be realized till then. The improvements in data technology saw the advent of data storage with time for which rest is history.

3. Scope of the Work

The current paper is a thorough survey and analysis of

new techniques, novel languages, improved and new algorithms for handling large data volume with different parameters. In this paper, gaps have been found between existing work and a novel algorithm has been proposed for function normalization. In the proposed algorithm, the data are pre-processed and a discrete decision table

Algorithm NIA_Norm

Function Generation::Function Normalization
Initialize R_{SET} // RSET is raw data set
Select $Data_{(Proc)}$ // $Data_{(Proc)}$ is data processing
If // $Data_{(Pre\ Proc)}$ is data pre processing
$Data_{(Proc)} = Data_{(Pre\text{-}Proc)}$
{
Create DDT //Discrete Decision Table for n attributes
For ($DDT_{(n)}$=1, DDT<$DDT_{(n)}$, $DDT_{(n)}$++)
{
RDT : Reduce Decision Table
}
For (RDT=1, RDT< $RDT_{(n)}$, $RDT_{(n)}$++)
{ // D_S Data set
Create D_S
}
Perform Normalization for D_S // Randomly generated Neuron threshold
For {each value of D_S under given value of threshold T}
(D_S=1, D_S< $D_{S\ AT\ T}$,D_S++)
{
CreateNN
Training of each NN
Perform testing
}
Else
Exit
Function Normalization (Parameter D_S, n)
Initialize $D_{SET\ (n)}$ and weights $W_{i,j}$ = 1/2p , 1/q for y= 0,1
Where p and q are the no. of negatives and positives respectively.
For (t = 1, t< T, t++) // i=1 to n
{
Normalize the weights for data set, $W_{t,,i\ =}\ w_{t,i}\ /\ \sum_{j=1}^{n} w_{t,j}$
Weight error of odd data set (D f(x, f , r, ϴ) is
$C_1 = \sum_{j=1}^{n} w_i\ |h_f(x_{i\ -}\ y_1)|$
} // where f_t, r_t, and θ_t are the minimizers of ε_t
For (C_1 =1, C≤C_h, C_1 ++)
{
Select D_{best} (best dataset)ε w_{error}
$\varepsilon_{t\ =}\ \min_{r, f, \theta} \sum_{i}^{n} v_i\ |h(x_i , f , r, \theta)|$or efficient implementation
}
If
{
$h_t(x)$!=0
select the value for $h_t(x) = h (x, f_{t,}, p_{t,} \theta)$
Update the $w_{t,,i}$
$C_{1(x)}$=1 or 0
}
Else
Exit

is created for performing the normalization of data sets by which one can compute the normalization function at any instant for all kinds of series of data. The selection of the data is based on the threshold value which is again user-defined. The flexibility and the robustness of the algorithm pertain to the user choice based scenario that is computed through the history or the genetic of the work done so far. One can implement the same through various utilities and tools as described in the paper in the previous sections.

4. Conclusion

In this paper the evolution of data has been discussed, tabular and conceptual description about tools and techniques of data mining has been included. Furthermore, the comparative study of distinct tools and techniques has been done which enables easy transformation of data into efficiently and effective. There are various methods available to improve efficiency of generating frequent item. Big data comes with its own difficulties in data capture, storage, searching, sharing, analysis and visualization. Several effective techniques and tools including machine learning, neural network, Genetic Algorithm and Artificial Intelligence are available to cater to the issues of data management and usage. However there is a scope of more innovation and creativity to optimize the whole process.

5. References

1. Manyika J, Chui M, Brown B, Bughin J, Dobbs R, Roxburgh C, Byers AH. Big data: The next frontier for innovation, competition, and productivity. USA: McKinsey Global Institute; 2011.
2. Herman H. An electric tabulating system. Berlin Heidelberg: Springer; 1982.
3. Chen CLP, Zhang C-Y. Data-intensive applications, challenges, techniques and technologies: A survey on big data. Information Sciences. 2014; 275(2):314–7. DOI: 10.1016/j.ins.2014.01.015.
4. Ramya AV, Sivasankar E. Distributed pattern matching and document analysis in big data using Hadoop MapReduce model. 2014 IEEE International Conference on Parallel, Distributed and Grid Computing (PDGC); Solan. 2014. p. 312–7. DOI: 10.1109/PDGC.2014.7030762.
5. Messick SJ, Tucker LR, Garrison HW. A punched card procedure for the method of successive intervals. ETS Research Bulletin Series. 1955; 1955(2):i–23. DOI: 10.1002/j.2333-8504.
6. Zaheer K, Anjum A, Kiani SL. Cloud based big data analytics for smart future cities. Proceedings of the 2013 IEEE/ACM 6th International Conference on Utility and Cloud Computing, IEEE Computer Society; 2013. p. 381–6. DOI: 10.1109/UCC.2013.77.
7. Hashem IAT, Yaqoob I, Anuar NB, Mokhtar S, Gani A, Khan SU. The rise of "big data" on cloud computing: Review and open research issues. Information Systems. 2015 Jan; 47: 98–115. DOI: 10.1016/j.is.2014.07.006.
8. Uskenbayeva R, Cho YI, Temirbolatova T, Kozhamzharova D. Integrating of data using the Hadoop and R. Procedia Computer Science. 2015; 56:145–9. DOI: 10.1016/j.procs.2015. 07 .187.
9. Li J, Roy P, Khan SU, Wang L, Bai Y. Data mining using clouds: An experimental implementation of apriori over mapreduce. 12th International Conference on Scalable Computing and Communications (ScalCom'13); 2012. p. 1–8.
10. Uzunkaya C, Ensari T, Kavurucu Y. Hadoop ecosystem and its analysis on tweets. Procedia-Social and Behavioral Sciences. 2015; 195:1890–7. DOI: 10.1016/j.sbspro.2015.06.429.
11. Kan HU, Cheung DW, Shaowei XIA. Adaptive interval configuration to enhance dynamic approach for mining association rules. Tsinghua Science and Technology. 1999; 4(1):1325–33.
12. Arora S Chana I. A survey of clustering techniques for big data analysis. 2014 IEEE 5th International Conference Confluence the Next Generation Information Technology Summit (Confluence); Noida. 2014. p. 59–65. DOI: 10.1109/CONFLUENCE.2014.6949256.
13. He H, Zhang WDZ, Chen A. Optimization strategy of Hadoop small file storage for big data in healthcare. The Journal of Supercomputing. 2015; 71(6):1–12. DOI: 10.1007/s11227-015-1462-4.
14. Cassales GW, Charao AS, Pinheiro MK, Souveyet C, Steffenel LA. Context-aware Scheduling for Apache Hadoop over pervasive environments. Procedia Computer Science. 2015; 52(10):202–9. DOI: 10.1016/j.procs.2015.05.058.
15. Assuncao MD, Calheiros RN, Bianchi S, Netto MAS, Buyya R. Big data computing and clouds: Trends and future directions. Journal of Parallel and Distributed Computing. 2015; 79-80:3–15. DOI: 10.1016/j.jpdc.2014.08.003.
16. Ozkose H, Ari ES, Gencer C. Yesterday, today and tomorrow of big data. Procedia-Social and Behavioral Sciences. 2015; 195:1042–50. DOI: 10.1016/j.sbspro.2015.06.147.
17. Subramaniyaswamy V, Vijayakumar V, Logesh R, Indragandhi V. Unstructured data analysis on big data using map reduce. Procedia Computer Science. 2015; 50:456–65. DOI: 10.1016/j.procs.2015.04.015.
18. Maitrey S, Jha CK. Handling big data efficiently by using map reduce technique. 2015 IEEE International Conference on Computational Intelligence and Communication Technology (CICT); Ghaziabad. 2015. p. 703–8. DOI: 10.1109/CICT.2015.140.

19. Barbierato E Gribaudo M, Iacono M. Performance evaluation of NoSQL big-data applications using multi-formalism models. Future Generation Computer Systems. 2014; 37:345–53. DOI: 10.1016/j.future.2013.12.036.
20. Lin JCW, Gan W, Hong T-P. Efficiently maintaining the fast updated sequential pattern trees with sequence deletion. Access IEEE. 2014; 2:1374–83. DOI: 10.1109/ACCESS.2014.2373433.
21. Wu X, Zhu X, Wu G-Q, Ding W. Data mining with big data. IEEE Transactions on Knowledge and Data Engineering. 2014; 26(1):97–107. DOI: 10.1109/TKDE.2013.109.
22. Kambatla K, Kollias G, Kumar V, Grama A. Trends in big data analytics. Journal of Parallel and Distributed Computing, 2014; 74(7):2561–73. DOI: 10.1016/j.jpdc.2014.01.003.
23. Hu H, Wen Y, Chua T-S, Li X. Toward scalable systems for big data analytics: A technology tutorial. Access IEEE. 2014; 2:652–87. DOI: 10.1109/ACCESS.2014.2332453.
24. Zhang Y, Ding S, Xu X, Zhao H, Xing W. An algorithm research for prediction of extreme learning machines based on rough sets. Journal of Computers. 2013; 8(5):1335–42. DOI: 10.4304/jcp.8.5.
25. Sandro F, D'Anca A, Palazzo C, Foster I, Williams DN, Aloisio G. Ophidia: Toward big data analytics for escience. Pocedia Computer Science. 2013; 18:2376–85. DOI: 10.1016/j.procs.2013.05.409.
26. Yunjuan L, Lijun Z, Lijuan M, Qinglin M. Research and application of information retrieval techniques in intelligent question answering system. 2011 IEEE 3rd International Conference on Computer Research and Development (ICCRD); Shanghai. 2011. p. 188–90.
27. Liu J Liu F, Ansari N. Monitoring and analyzing big traffic data of a large-scale cellular network with Hadoop. IEEE Network. 2014; 28(4):32–9.
28. Rajalakshmi V, Mala GSA. Anonymization by data relocation using sub-clustering for privacy preserving data mining. Indian Journal of Science and Technology. 2014 Jul; 7(7).
29. Kyoo-Sung N, Doo-Sik L. Bigdata platform design and implementation model. Indian Journal of Science and Technology. 2015 Aug; 8(18). DOI: 10.17485/ijst/2015/v8i18/75864.
30. Vijaykumar S, Saravanakumar SG, Balamurugan M. Unique sense: Smart computing prototype for industry 4.0 revolution with IOT and bigdata implementation model. Indian Journal of Science and Technology. 2015 Dec; 8(35). DOI: 10.17485/ijst/2015/v8i35/86698.
31. Park HW, Yeo IY, Jang H, Kim NG. Study on the impact of big data traffic caused by the unstable routing protocol. Indian Journal of Science and Technology. 2015 Mar; 8(S5). DOI: 10.17485/ijst/2015/v8iS5/61480.
32. Yasodha P, Ananthanarayanan NR. Analysing big data to build knowledge based system for early detection of ovarian cancer. Indian Journal of Science and Technology. 2015 Jul; 8(14). DOI: 10.17485/ijst/2015/v8i14/65745.
33. Kim KW, Park WJ, Park ST. A study on plan to improve illegal parking using big data. Indian Journal of Science and Technology. 2015 Sep; 8(21). DOI: 10.17485/ijst/2015/v8i21/78274.
34. Dhamodaran S, Sachin KR, Kumar R. Big data implementation of natural disaster monitoring and alerting system in real time social network using Hadoop technology. Indian Journal of Science and Technology. 2015 Sep; 8(22). DOI: 10.17485/ijst/2015/v8i22/79102.
35. Kim BS, Kim DY, Kim KW, Park ST. The improvement plan for fire response time using big data. Indian Journal of Science and Technology. 2015 Sep; 8(23). DOI: 10.17485/ijst/2015/v8i23/79198.
36. Karthick N, Kalarani XA. An improved method for handling and extracting useful information from big data. Indian Journal of Science and Technology. 2015 Dec; 8(33). DOI: 10.17485/ijst/2015/v8i33/60744.
37. Somasekhar G, Karthikeyan K. The pre big data matching redundancy avoidance algorithm with mapreduce. Indian Journal of Science and Technology. 2015 Dec; 8(33). DOI: 10.17485/ijst/2015/v8i33/77477.
38. Lakshmi M, Sowmya K. Sensitivity analysis for safe grain-storage using big data. Indian Journal of Science and Technology. 2015 Apr; 8(S7). DOI: 10.17485/ijst/2015/v8iS7/71225.
39. Mamlouk L, Segard O. Big data and intrusiveness: Marketing issues. Indian Journal of Science and Technology. 2015 Feb; 8(S4). DOI: 10.17485/ijst/2015/v8iS4/71219.
40. Noh K-S. Plan for vitalisation of application of big data for e-learning in South Korea. Indian Journal of Science and Technology. 2015 Mar; 8(S5). DOI: 10.17485/ijst/2015/v8iS5/62034.

Appendix

A_{CA}- Intelligent Automatic Caching of Data, DV_{Vz}- Discovery Visualization, FLY_{U}- A User-Friendly Graphical Interface LRN_{EVO}- Evolutionary Learning, D_{EX} -Data Exploration, OT_{DE} - Outlier detection, C_{LT} – Clustering, TX_{M} - Text Mining, TS_{AY} - Time Series Analysis, SN_{AY}- Social Network Analysis, PL_{CP}- Parallel Computing, GP_{VZ}- Graphics Visualization, BG_{Wapps} - Web Application Big data. E_{H-} Error Handling, A_{LA-} array language, $IF_{CSV/EXC/R}$ - Support input file csv txt/excel/RData File, $STA_{MIN/MAX}$ - Support Statistics: Min/Max, $MDL_{DT/RF/LR}$ - Support Modeling: Decision Trees/Random Forests/Logistic Regression, $EVA_{CM/RC/CC}$ - Support Evaluation: Confusion Matrix/Risk Charts/Cost curve, $CHT_{BP/HG/CR/DG}$ - Support Charts: Box Plot/Histogram/ Correlations/Dendrograms, T_{RF} – Transformations, B_{DEB}- Better debugger, SH_{SC} - Shortest scripts, S_{NVE} - Suitable for no voice Experts, $S_{VZ/STA/AS/OD/PO}$ - Suitable for Visualization/Statistical/ Attribute Selection/Outlier detection,/parameter optimization. MO_{AY} - Molecular analysis, MS_{SPY} - Mass spectrometry , CY_{DVK} - Chemistry Development kit , EV_{ALGO}-Evolutionary algorithms, Fz_{ST}- fuzzy systems, PY_{STA} - Purely statistical R_{O}-Robust, R_{E}-Reliable, PIC_{RP} -

Representation is easy with pictures, MA_{GUI} - GUI as great as memory aids, ES_{U} - Ease of use, RM_{EX} - can be extended in RM, BG_{INS}-Big installation, $RPcp_{L}$-Limited reporting capabilities

DB_{H}-Requires prominent knowledge of database handling, EMT_{L} - Limited error measurements, OZP_{PR} poor parameter optimization ,L_{ALGO} - Limited algorithms, $L_{DM/ALAK}$- Less specialized for data mining and less knowledge of array language , DOC_{PR}-Poor documentation, STA_{WK} - weak classical statistics, CR_{WK}- weak csv reader , PIC_{L}- Pictures can be a limited Representation, H_{RDLY} - High readability, OOP_{ES} - Easy object oriented paradigm, ES_{EX}- easily extensible, UC_{SG}- Strong Unicode support, LIB_{POW}- powerful standard library, TKZ_{TX} - Tokenizing the text, CT_{WD} - Counting the remaining words, ENV_{DS}- works well in distributed environment, DM_{BG}- Doing data mining tasks on Big data, BG_{AYQ}-Analyze Big data quickly, $IC_{MX/VT}$- include matrix and vector libraries, FIF_{DS}- Support distributed fitness function, M_{PTR}- pattern mining, U_{IM}- user interest model, S_{Y}-scalability, NO_{ALGO}-No tuning or algorithm choices needed, R_{FE}&H_{BE} -Integrated real-time front-end & batch Hadoop-based backend, OH_{BE} - Offline Hadoop backend are not integrated. STA_{CAL}- Includes packages for statistical calculations. $D_{2D/3D}$- 2D and 3D interactive visualization of data, $F_{HT/CH/AY/CAY/NN/FZ}$- functions of histograms/charts/analytic calculations/ neural networks/Cluster analysis/Fuzzy. VZ_{EXF} - Data stored in external files for convenient visualization. $F_{MTH/NCAL/VZ}$ - Provide Mathematics functions/numeric calculations/visualization, $W_{STA/ARI}$- provide better work for statistics/artificial intelligence, $B_{NS/MAY}$ - Better work for natural science and market analysis. LKJ_{Y}- Support Jython &Lack of Knowledge, $CLS_{LDA/LR}$ - Classification: Linear discriminant analysis/logistic regression, $REG_{LSQ/RR}$ - Regression: least squares/ridge regression, $CLU_{HC/KM}$ - Clustering: hierarchical clustering, k-means, DMR_{FDAY} - Dimensionality reduction: Fisher discriminate analysis, $SP_{VM/PLS/MHC}$ - Support vector machine, partial least squares, memory saving hierarchal clustering, DS_{H} - handling series of dataset, EQ_{CT} - equality constraints, D_{MU} - data manipulation, AT_{SCP} - automation of common tasks with scripts, $WLSQ_{LAGM/NM/GA}$- support three weighted least squares methods: Levenberg -Marquardt algorithm / Nelder-Mead method/ Genetic algorithm, $GP_{PLS/PRS}$ Fill gap b/w plotting software/programs specific, DS_{MPH} - won't allow you to measure the phasing of the dataset, AN_{NIE} - A Nearly-New Information Extraction system, GUI_{JB} - Re-task able components (Java beans), including GUI components, LE_{D}- benefit LE system developer, $EXO_{MAT/DBT}$ - Ability to exploit the maturity/efficiency of database technology, E_{BDC} - Easy modeling of blackboard-type distributed control regimes, Q_{E}-Quantitative evaluation, R_{Y} - Repeatability, ARC_{DEG} - Architecture development at same time as word sense disambiguation engine, OC_{DDS} - Distributed data storage in Oracle or PostgreSQL (over JDBC), DB_{SW} - Speed of database i.e. Tipster database slow for large amounts of lexical data.

E_{IM}-Easy to implement, LI_{PY}-Uses large item set property, E_{PRL}-Easily parallelized, SAP_{AO}- Apriori Algo. can be very slow, TR_{DB}-Assumes Transaction database is memory resident, SC_{DB}-Requires many database scans, CL_{AG}- Items automatically assigned to clusters, $RY_{SC/SI}$-Relatively scalable/simple, CSC_{DS}- Suitable for datasets with compacts spherical clusters, SCK_{M}-k means relatively scalable in processing large data sets, FK_{M}- k mean is fast, $HS_{PH/NO/OU}$-High sensitivity to initialization phase/noise/ outliers, FE_{LO}- Frequent entrapments into local optima, $NC_{VS/DY}$-Inability to deal with non-convex clusters of varying size/density, ODK_{M} -K-means applicable only when mean objects defined, $H_{NOD/OU}$-k-means unable to handle noisy data and outlier, $DT_{UD/IP}$ - Decision trees are simple to understand/ interpret. , $H_{NM/CD}$ -Handle both numerical and categorical data, ATC_{LS}- Good classification provide definitive array of types, R_{CX}-Reduce complexity, SLR_{Z}-Allow us to recognize similarities, DT_{NV}- Decision Trees Hard to run for numerical values , $SKR_{XV/SW}$: SVM not look effective without using kernel but kernel makes it computationally expensive/ slow, LR_{BM}-Logistic Regression Lack of benchmark results, UNB_{Y}- Naive Bayes Works strangely on unbalanced classes, $PD_{SH/LG}$-Accurate prediction for short/long-term, WA_{PP}-Wide scope of application, $PZ_{DF/HD}$-Difficult and hard to popularize, D_{EX}- excessive data as required, K_{B}-kilobyte, M_{B}-megabyte, G_{B}-gigabyte, T_{B}-terabyte, P_{B}-petabyte, E_{B}-Exabyte, Z_{B}- zettabyte, Y_{B}-yottabyte, NK_{AO} -K Nearest Neighbor Algo, NB_{Y}- Native Bayes, DC_{TE} -Decision Tree, SP_{BA}-Statistical Procedure based approach, ML_{BA}- Machine learning Based approach, OU_{ANA}-Outlier analysis, EV_{ANA}- Evolution Analysis, G_{A}- Genetic Algo., R_{SA}-Rough set approach, FZ_{SA}-Fuzzy set approach, RG_{EG}- Regression, GL_{MD}- Generalized Linear Models, V_{M}-Vector Machine, LR_{EG}- linear regression, PR_{EG}-Polynomial Regression, LGR_{EG}- Logistic regression, $GUI_{EXDA/EXP/KF}$ - Support graphical user interfaces of exploratory data analysis/The Experimenter/

the Knowledge Flow, $BS_{LR/PR}$ - Bootstrap tool for creating new Language Resources and Processing Resources, C_{LIB} - provide class library that implements the architecture, FK_{GDE} - graphical development environment built on the framework, V_Z-Visualization, D_{AT}-Data Analytics, I_{TC}-Interaction, L_{TX}-Large toolbox, S_{ITF}-Scripting interface, EX_{DOC}- Extendable Documentation, GUI_{IV}-Intuitive GUI, B_{EX} -Batch executions, D_M-data mining, P_Y-Python, D_H/A_F- Data handling/new aggregation functions, MA_{CV}-Macro viewer shows macros values, F_{OD}- File operators operate directly, SM_{FLT} - Spam filters, S_{Y-}Scalability, I_{UIF}-Intuitive user interface, H_{EX}-High extensibility, B_{EX} -Batch executions, $WF_{IMP/EXP,}$ - Import/export of workflows, P_{EX}-Parallel execution on multi-core systems, API_{PL} - well-defined API for plugin extensions, CLS_{DV-} Classification Discovery, CLU_{DV-} Cluster Discovery, REG_{DV-} Regression Discovery, PA_{LM}-Poor absence of local-minima.