International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-2, July 2019

Techniques for Data Extraction from Heterogeneous Sources with Data Security

Kimmi Kumari, M Mrunalini

Abstract*: Data Extraction is the process of mining or fetching relevant information from unstructured data or the heterogeneous sources of data. This paper aims at mining data from three different sources such as online website, flat files and database and the extracted data are even analyzed in terms of precisions, recall and accuracy. In the environment of heterogeneous sources of data, data extraction is one of the crucial issue and therefore considering the present scenario, we can observe that the heterogeneity is expanding widespread. So this paper focus on the different sources for the data extraction and provides a single framework to perform the required tasks. In this paper, healthcare data are considered in order to show the processing starting from data extraction using three different sources to dividing them in to two clusters based on the thresholds value which has been calculated using cosine similarity and finally calculations of parameters like precisions, recall and accuracy for analyzation purpose. Fetching data online is the task in which we cannot fetch simple string from any website. The backend of each page is html and hence this paper focus on extracting that html of the page while mining data from any web server. The webpage contains a lot of html tags and all of these cannot be removed because they are complex tags which cannot be removed by regular expressions. But still 60% filtered data can be attained as demonstrated in this paper as most of the waste html will be removed. While filtration of the data, we should also note that the content containing Google APIs cannot be removed. So filtered data will contain the contents and tags which does not contain Google APIs. In order to provide data security while extraction, the connection string is being used to avoid tampering of data. This paper also focuses on one of the arguable concepts present in the generation of big data which is Data Lake. In originality, the origin about the idea of Data Lake appears from the field of business. An architectural approach which is specially designed in order to store all the data which are potentially relevant in a repository located centrally is referred to as Data Lake. The data which are stored in the central based repository are fetched from the sources belonging to public as well as enterprises and these data are further used for the purpose of organization, discovery of hidden facts, understanding of new concepts, analyzation of stored information etc. Many challenges and concerns related to privacy are faced during the adoption of Data Lake as it is a new concept which brings revolutionization. This paper also highlights some of the issues imposed by Data Lake.

Keywords-Accuracy, Data Extraction, Data Lake, Data Security.

Revised Manuscript Received on 30 July 2019.

***** Correspondence Author

Kimmi Kumari*, MCA, M S Ramaiah Institute of Technology, Bangalore, India.

Dr. M mrunalini, MCA, M S Ramaiah Institute of Technology, Bangalore, India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license http://creativecommons.org/licenses/by-nc-nd/4.0/

I. INTRODUCTION

The continuous challenge for healthcare professionals is the retrieval and extraction of medical data and this is due to the scarcity of technologies integration as well as important data which are time consuming and even involves manual workflows [1]. The data inaccessibility renders its inconsequential even though the data exists. This may result in redundancy of data by duplicating medical tests and studies which are really not necessary but happens to be available at the time of care. In the absence of complete and definitive data, medical providers renders the diagnostic and treatment decisions [2]. Time delays, diminished diagnostic confidence, excessive unnecessary consultations etc. are some of the adverse outcomes of the medical errors and inaccessible or incomplete medical data [3]. The Volume and the complexity of the medical data is increasing exponentially due to which the current state of inefficient data delivery are magnified at the time of requirements. There are various computerized data mining and applications for decision support which provides many benefits related to theory in order to enhance data extraction and comprehension. But still, the operation without the full complement of accessible data faced a lot of difficulties [4]. The most important fact is that the key data elements should be present in the computerized data analysis in order to avoid faulty and incomplete analysis. The best solution for this problem is computerized data mining which takes care of quality and completeness of input data [5]. This paper proposes the techniques in order to analysis the incompleteness of data by calculating precision, recall and accuracy. Processing of huge amount of healthcare data will take huge time so for this instance, the paper shows processing of a small segment of healthcare data. This paper also have the provisions to extract the ASCII values of the source data provided which doesn't have the unnecessary html tags. This ASCII values are used to calculate the threshold values and based on these thresholds which is calculated based on the cosine similarity, clusters are created. Thresholds are the first connections which changes with the change in similarity values. For examples, suppose we have 4 connections 1->2, 1-> 3, 1->4, and 1 -> 5. Now the first threshold will be 1->2 cosine similarity value as we are starting with 1 and 2 and furthermore it changes as more documents are added. K-means algorithms are used for cluster creation. Cluster creation is important because collecting data can never be the solutions to security aspects. Data management is the only way to organize the data so that when it comes to retrieval we will have significant reference point to look into.

Examples for these can be University Organization like Google, Yahoo etc. The characterization of big data can be done using a particular strategy to extract certain insights from a huge amount of data which can be structured as well as unstructured in nature at a faster rate [6]. Approaches which are traditional in nature cannot handle the data which has been captured with its insights being fetched but this can be fulfilled by enabling the big data because it helps in increasing the worth of the insights [8]. Hadoop and spark are the new technologies which are highly demanded as they serve as the good example of enabling big data. The reason behind the increasing popularity of these technologies which enable big data is that it support storing the data in a distributing and scale out manner. They are also used in the generic hardware as they process the resources contained in it. The discussion of big data also includes an interesting concept related to it i.e. Data Lake [9]. The idea behind data lake invention was not restricted only to handle great diversity of data or generation of high scales of data with respect to speed but to help the field of analytics by providing them the great range of versatility and agility which can help in empowering the younger worker with interested knowledge in an enterprise. The best example for this is the services provided by cloud as they are tend to provide the same level of flexibility but the economics of scale are quite good which is beneficial in enabling data lakes[10]. There are some technologies existing in the current era which business normally uses to provide functions related to analytics. Some of these technologies includes Relational databases, Data warehouse related to enterprise, Applications used in BI and data marts etc. Apart of all the advantages mentioned above, there are a lot of restrictions imposed by these legacy architectures. Traditional approaches faces some common challenges related to limitations in scale and accessibility, huge increment in cost, gaps in the performance, designs which are rigid in nature. Most of the organizations must first prepare use case which are specific and clear followed by implementation of work in a backward manner which can be done using a fixed model related to analytics containing raw data in it which has been fetched from a particular kind of source. Further the transformation of data can take place based on narrow definitions and lastly a hardware platform is used for the loading purpose. An attempt to bring any changes in this system results in the increment of the difficulty level as well as makes the process time-consuming. Clients which are associated to business are restricted to only one specific metrics or reports which are existing and they have started building an assumptions that it is of no worth expecting something new from the IT professional which often results in hindering their creativity. Due to this reason, huge amount of data are discarded and even ignored. Big data with Data Lake targets this whole insights and focus on turning their workflow. The environment should be built in such a way that the data which are discovered should be accessible to everyone that have an idea or any questions. The flexibility should be given in order to analyze the discovered data in whatever way they want to fulfill their requirements.The organization of this paper are as follows: Section 1 of this paper discusses about the concepts of Data Lake in big data generation along with the techniques being implemented for the data extraction from heterogeneous sources. Section 2 describes the techniques used in generating the values from the source data. Section 3 highlights the calculations pattern of initializing threshold and cluster division followed by section 4 which explains the parameters that has been used for the analyzation purpose. Last section of this paper discusses the techniques handled to fetch data from database/text files followed by the challenges imposed by Data Lake along with its primary concerns and conclusion with future directions of the paper.

Fig 1 shows the design of the single framework that has being developed in order to perform fetching of the data from the heterogeneous sources that incudes database, online website and flat/text files.

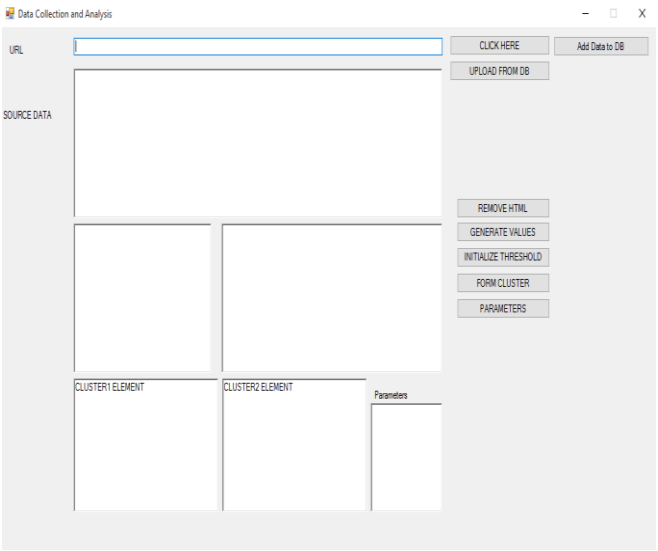

Fig 1. Design framework for data collection and analysis.

II. TECHNIQUES FOR DATA EXTRACTION FROM HETEROGENEOUS SOURCES AND THEIR VALUE GENERATION

String processing is never supported in any context as a normal string value contains ambiguities. Suppose I want to write Simmi but I have written $immi, it is different for a computer. Here a numeral system is always good to use in order to remove ambiguities. The values as shown in screenshot 4 are generated using ASCII interpretation of the characters. The tool takes data from three sources. One from online portal, second from databases and third from text files. Fig 2 explains the working of the online data extraction and its processing. A medical website i.e. www.fortishealthcare.com is uploaded in the URL text box and the tool extracts the html part. The html part is cleaned up with regular expression and further the ASCII encoding of the data is done. The data is bifurcated into paragraphs and the similarity index is calculated. Based on the similarity values, k means clustering is applied and two segregated clusters are created. Evaluation parameters are computed as precision and recall along with f measure.

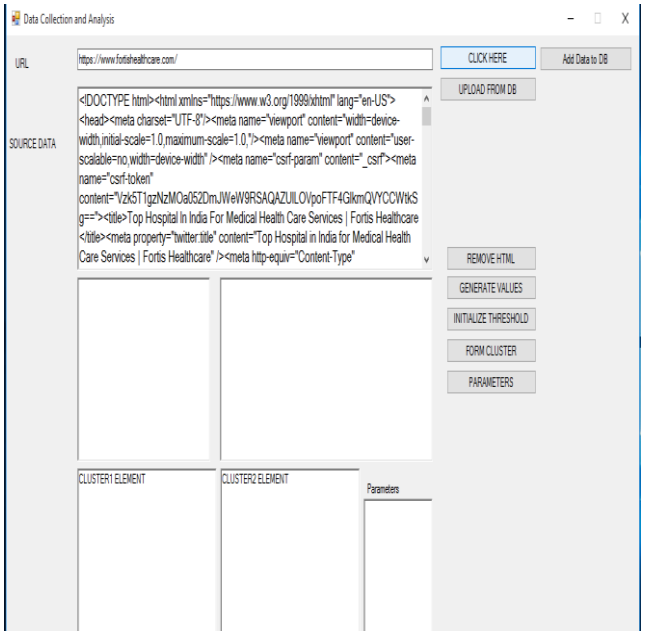

Fig 2. HTML Part Extraction

Fig 3 contains the source data after the html part of it is being removed. As mentioned before except the contents which have Google's API, the tool removes all the html tags included in the contents of the source data.

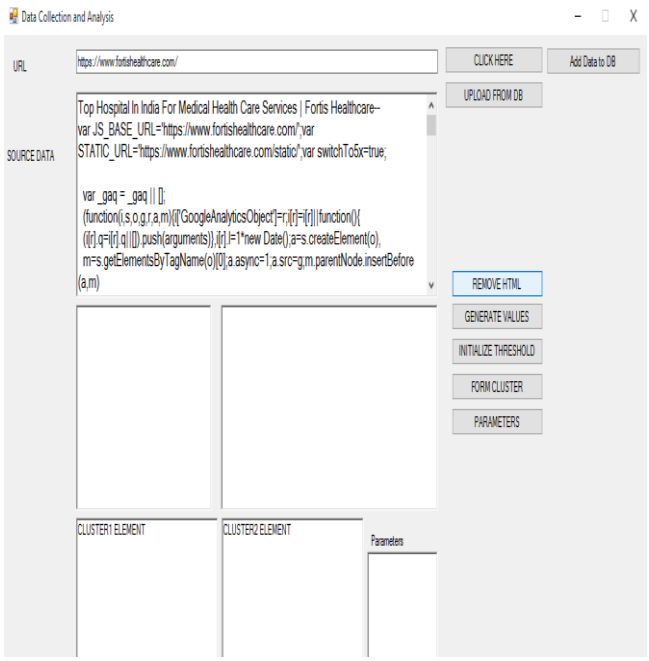

Fig 3. Removing HTML Tags

In Fig 4, the values are generated based on the source data whose content does not contain html tags. For Example, Top is the starting word being generated while fetching process so the ASCII value of T should be added to the ASCII value of o and p and the total sum i.e. 84+111+112= 307 is the result of the values being calculated as seen in screenshot 4. Same applies for the other words in the source data column.

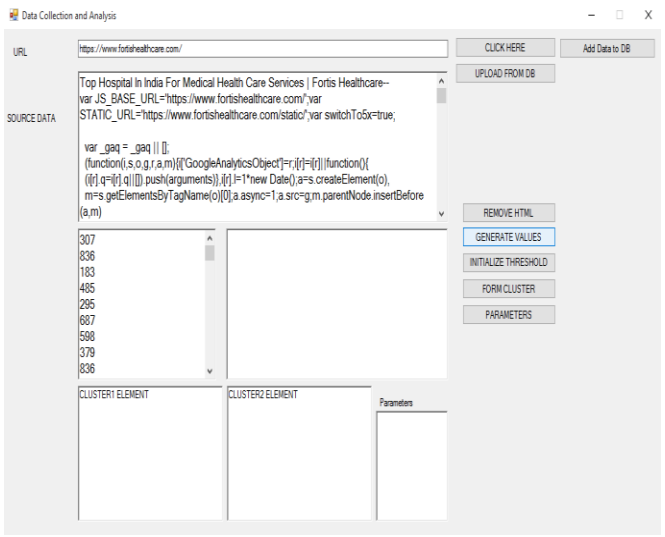

Fig 4. Values generation after removing Html tags

III. CALCULATION PATTERN FOR THRESHOLD INITIALIZATIONS AND CLUSTER DIVISIONS

Based on the ASCII values being generated from the source data, Thresholds value will be initialized and calculated using the similarity which can be defined by using the angle between the two vectors or by cosine angle. The measure of the cosine is given by the formula as proposed in the authors in [7] as $s^{\copyright}(x_a + x_b) = (x_a + x_b)/ |x_a|^2 .|x_b|^2$ which helps in capturing the similarity in terms of invariant understanding in scales. One of the important and strongest property delivered by cosine similarity is that it is not dependent on the length which is $s^{\copyright}(z\, x_a + x_b) = s^{\copyright}(x_a + x_b)$ where $z > 0$. The advantage of this is the documents which includes same compositions but the totals contained in that varies are treated in an identical manner. Due to this reason, the demand for cosine similarity measure is increased for text documents. This property also is used for normalization factor for more and efficient processing.

In Fig 5, threshold column contain the first statement as Main word Value 0 connected word value 4 that means 0^{th} value is 307 which is present in the ASCII column and 4^{th} value is 295. The matrix will be (307, 1) and (295, 2) and therefore the distance calculated are as follows:

$$\text{dist.} = \text{sqrt}((307-295)^2 + (1-2)^2)$$
$$\text{dist.} = \text{sqrt}((12)^2 + (-1)^2)$$
$$\text{dist.} = \text{sqrt}(144 + 1)$$
$$\text{dist.} = \text{sqrt}(145)$$
$$\text{dist.} = 41.76122$$

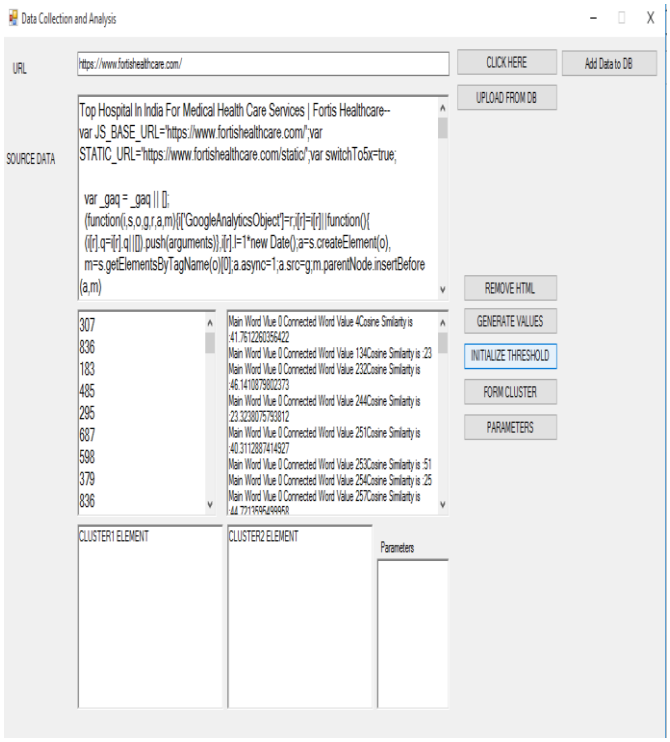

Fig 5. Threshold initializations

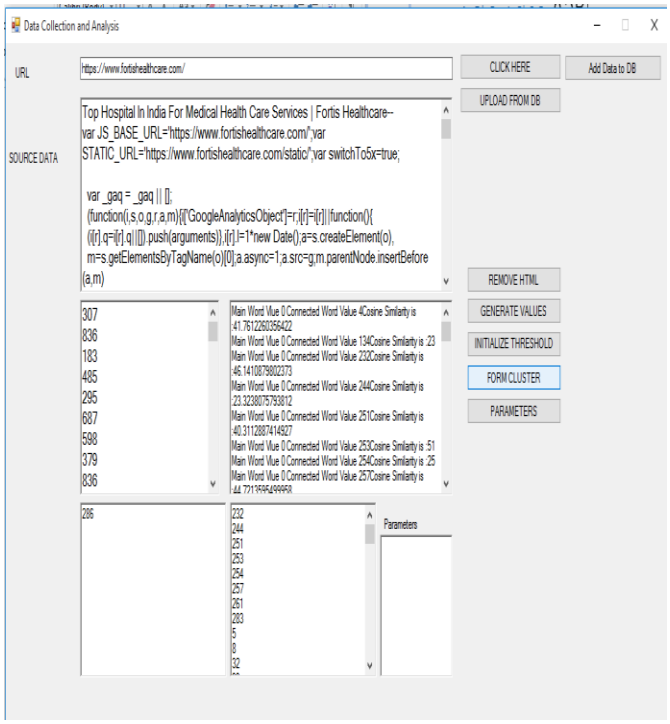

Fig 6. Cluster divisions

After the cosine similarity calculations, K-means techniques are used in order to divide the data into clusters with the focus of making the management of data easy. A current value has been calculated using the values of cosines and the current value less than 0 are put in one cluster and the current value greater than 0 are included in the other cluster. In Fig 6, we can view two clusters which includes the ASCII values in it based on the calculations of the current value. Below is the part of code which shows the current value calculations and the implementation of the k means techniques.

// datapoint [0, 0] = datavalue[i]; // Initialization of datavalue for its i^{th} iterations.

```
// double currentvalue = (double) (datavalue[i]); //declaration of the current value.
    if (currentvalue != 0)
      {
        double margin = currentvalue - (currentvalue * 80 / 100);  // margin calculation.
            if (diff <= margin)
            {
              try
              {
                Kmeanmapping [kmeanlength, 0] = i; // K-means implementation.
                Kmeanmapping [kmeanlength, 1] = j;
                Kmappedvalue [kmeanlength] = diff;
               // mapcount = mapcount + 1;
                kmeanlength = kmeanlength + 1;
              }
              Catch {}
```

IV. PARAMETERS FOR ANALYZATION PURPOSE

When we are dealing with the retrieval of information then precision and recall plays a vital role. Precision are those which are referred to as the fraction of retrieved documents that are relevant to the query and recall is the fraction of relevant documents that are successfully retrieved [12]. For example, for a text search on a set of documents, precision is the number of correct results divided by the number of all returned results [13]. Precision takes all retrieved documents into account, but it can also be evaluated at a given cut-off rank, considering only the topmost results returned by the system. This measure is called precision at n or P@n. Precision is used with recall, the percent of all relevant documents that is returned by the search. The two measures are sometimes used together in the F1 Score (or f-measure) to provide a single measurement for a system [11].

For a text search on a set of documents, recall is the number of correct results divided by the number of results that should have been returned [14]. In binary classification, recall is called sensitivity. It can be viewed as the probability that a relevant document is retrieved by the query. It is trivial to achieve recall of 100% by returning all documents in response to any query. Therefore, recall alone is not enough but one needs to measure the number of non-relevant documents also, for example by also computing the precision.

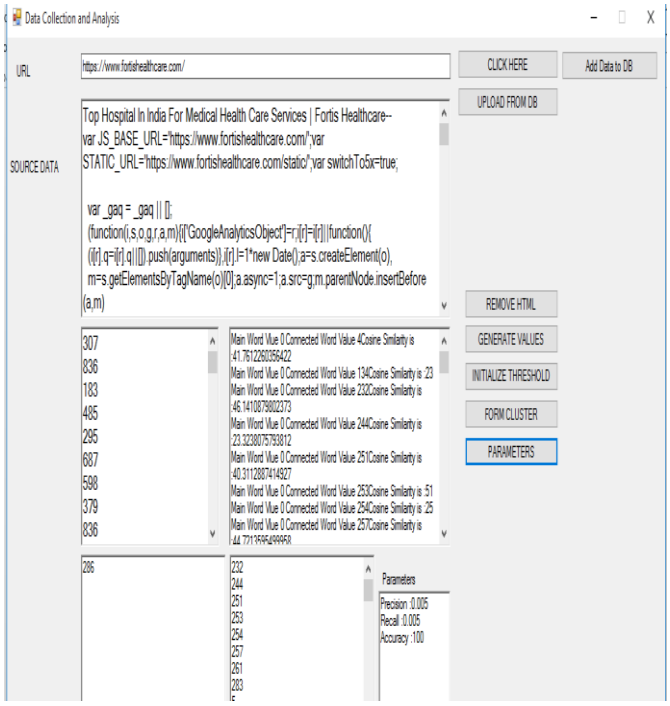

Fig 7. Parameters calculation

Fig 7 shows precision value as 0.005 which means that 0.5% retrieved documents relevant to the query is fetched and recall value as 0.005 which means that 0.5% of the relevant documents are successfully retrieved. Accuracy is also calculated for the analysis purpose as 100%. Accuracy does not perform well with imbalanced data sets. For example, if you have 95 negative and 5 positive samples, classifying all as negative gives 0.95 accuracy score. Balanced Accuracy overcomes this problem, by normalizing true positive and true negative predictions by the number of positive and negative samples, respectively, and divides their sum into two [17]. Regarding the previous example (95 negative and 5 positive samples), classifying all as negative gives 0.5 balanced accuracy score out of the maximum balanced accuracy one, which is equivalent to the expected value of a random guess of a balanced data [15].

Balanced Accuracy is suggested to use to measure how accurate is the overall performance of a model is, considering both positive and negative classes without worrying about the imbalance of a data set [16]. Since most of the real data sets are imbalanced, Balanced Accuracy metric is suggested instead of Accuracy metric. Additionally, the predicted positive condition rate (PPCR) identifies the percentage of the total population that is flagged; for example, for a search engine returning 30 results (retrieved documents) out of 1,000,000 documents, the PPCR is 0.003%.

V. TECHNIQUES FOR DATA EXTRACTIONS FROM DATABASE AND TEXT FILES

This section of the paper clearly shows that the framework which is being developed also have the provisions to either upload the data from the database as well as from the text file. Before uploading the contents of the database or the flat files, the creation of their content should already be present. Fig 8 shows that the connection is being established once the option for add data to DB is selected. Database is created with the name as "weth" and the table created under the given database is "wethtab" as shown in screenshot 11. While dealing with the data extraction part from the database, first step included is to obtain a secure connection using a connection string in the same format as defined below:

```
<connectionStrings>
  <add name="kim" connectionString="Data Source=DESKTOP-EVH27LF\MSSQLSERVER01; database=weth; Integrated Security=true" />
 </connectionStrings>
```

In the above connection string, Source and Database created with the name "weth" is been clearly specified. By doing so, we can provide security to our important and confidential data as every time we logged in and wants to fetch information from the database, authentications for that particular user will be done through this connection string.

```
con.ConnectionString = ConfigurationManager. ConnectionStrings ["kim"].ConnectionString;
```

The above syntax even requires the same name i.e. "kim" to be passed as the parameter while calling the connection string for the configuration purpose.

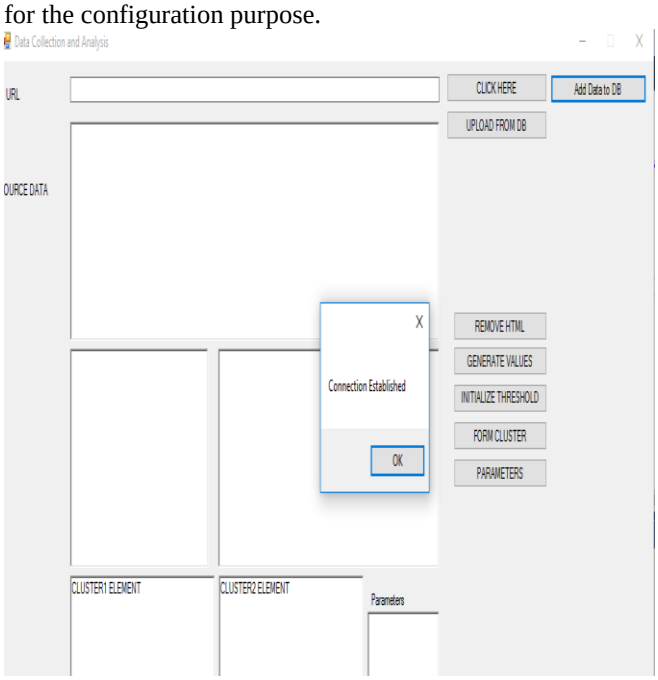

Fig 8. Connection with the database is performed

Fig 9 shows the window which appears when the connection to the database is completed successfully. This window gives the flexibility to upload the content of the text files present in your local computer or even copy pasting options are also supported as the user wish to create its own content. The same content uploaded from the text files can be saved to the database by selecting the "SAVE TO DB" option.

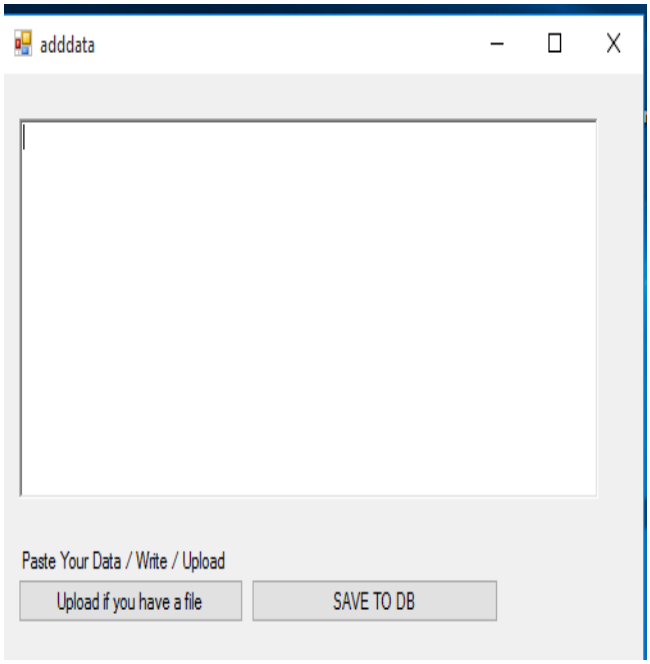

Fig 9. Window for uploading contents of text files

In Fig 10, the contents from the flat/text files are uploaded which was initially created in the local computer. After saving the same content to the database, the results can be viewed by selecting the table of the database i.e. weth.

SqlCommand cmd = new SqlCommand ("insert into wethtab values (@id, @data)", con);

The above syntax shows the table "wethtab" of the database "weth" contains two fields i.e. id and data. Precautions should be taken to use the correct table in the above syntax to avoid errors.

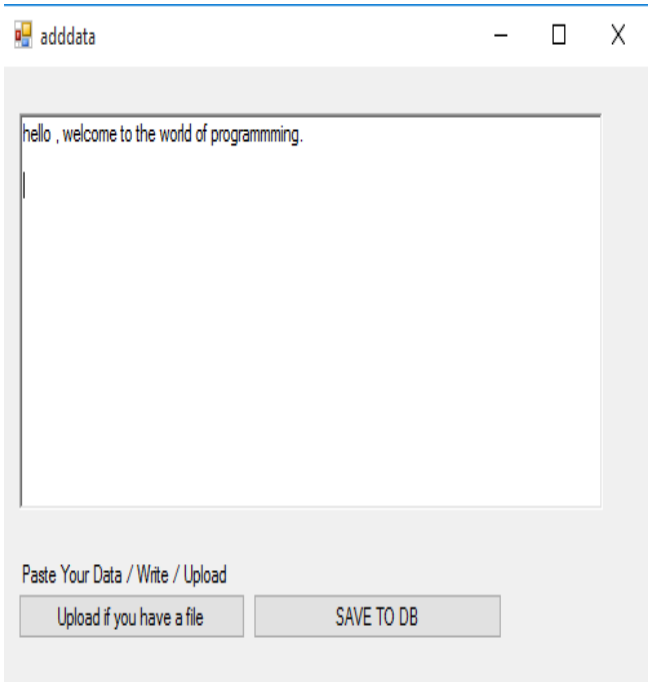

Fig 10. Contents of text files are uploaded

Fig 11 shows the window for the database creation and the same data which was uploaded from flat/text file are successfully inserted in the table "wethtab" which has been used for the insertion purpose. There is no restrictions in the creation of the database and tables so it provides flexibility to the users.

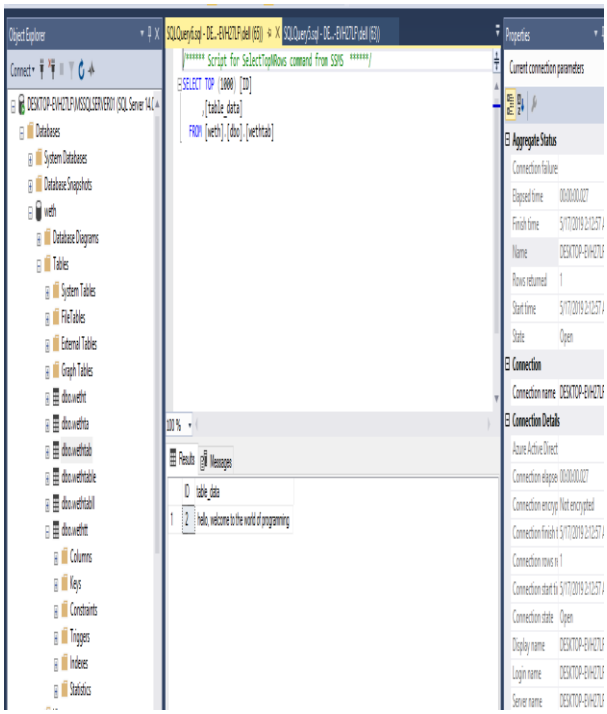

Fig 11. Glimpse of the Contents when added to the database

VI. CHALLENGES AND PRIMARY CONCERNS OF DATA LAKE

There are a lot of primary concerns imposed by Data Lake in business point of view as well as technical point of view. When considering the business aspect, there are chances for Data Lake to become another hype for marketing in the area of hadoop. Agility and accessibility are the major need for data analysis which leads to the emergence of Data Lake. Data Lake can easily be the means of providing solutions to different platforms but point to note is that it cannot be the solutions as expected by the various enterprises. Since frameworks like Sparks, Flink etc. are used for the implementation of current data lakes, the news of marketing hype is rising rapidly [18]. As far as technical aspect is considered, Data Swamps can easily take the form of Data Lake. One of the pitfalls of Data Lake also noticed by their supporters are that no one knows what kind of data has been put into the Lake and so the chances of incorrect data, repeated data are increased. The guarantee is also not provided by the data in the lake for its veracity [19]. There is no prevention being taken till now to overcome this issue. Just imagine, if nobody knows about the data contained in the lake then how anybody can determine whether the data's are corrupted or not. Fig 12 shows a simplified view of Data Lake which has the ability to accept all varieties of data in it without the need for oversight and asking for governance. This is one of the reason of analyzing the data from time to time. Data Lakes focus more on storing the data rather than determining why and how those data's are utilized or being secured or governed [20].

It is very difficult to gain values from Data Lakes as there is no unique identifiers who can perform advance search of huge data volume.

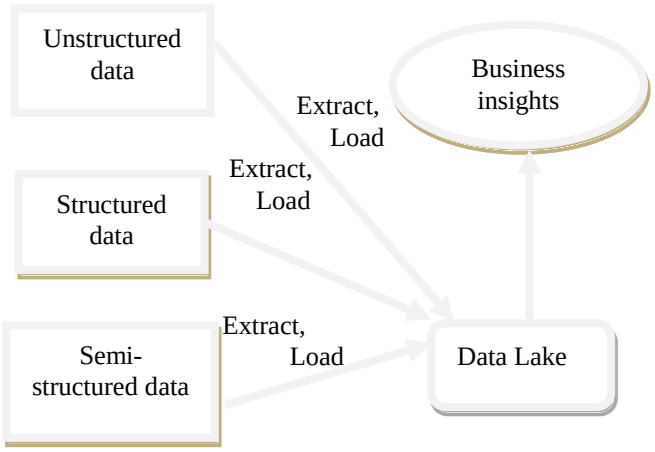

Fig 12. Simplified view of Data Lake

VII. CONCLUSION AND FUTURE DIRECTIONS

We use Data Lakes in order to solve two problems. One is the old problems i.e. data silos and the new problem i.e. the challenges and issues which are imposed by initiatives of big data. The data's are stored all together in Data Lake rather than handling them independently so that old problems can be overcome. The challenges of big data era is easily handled by the new problems such as big data impose some challenges related to 5 V's like velocity, veracity, value, volume and variety are tried to be resolved by the concept of Data Lake. The reason of data silos to be likely to occur is when the data being generated from different departments but same organizations are restricted only in their stores of data. Now where the role of data lakes comes into picture when it tries to integrate the data from different sources into a single place so as to reduce the occurrence of data silos. Traditional data are inefficient to handle huge volume of data coming from various source so in the context of big data, in order to overcome data silos, data lakes are useful in terms of tackling with variety and volume of data. There is no restrictions imposed by Data Lake on the structure of data as well as volume of data. Any data can be easily accepted and stored in Data Lake. When variety of data's are retrieved from heterogeneous sources then the need for data pipelines increases because it is beneficial in processing high velocity data processing. The attention is required to be paid by the data pipelines on the request given by V's as they are built in order to feed data into the lakes. The metadata with the extra data needed for future use along with the data extracted are carried by the data pipelines. Data veracity cannot be assured as the security and management of the data is weak in Data Lake. So work towards this issue should be taken care. This paper also focus on the implementations of techniques in data extraction from heterogeneous sources along with the discussions of Data Lakes. The importance of parameters like precisions, recall and accuracy are listed out which is further used for the analyzation purpose. K means techniques are being used for the cluster divisions and in order to initialize thresholds, cosine similarity is being used. These techniques are being briefly discussed and their implementation on the source data has been presented. Healthcare data are considered for data extraction purpose and all the mentioned techniques are worked on those source data. A lot of screenshots of their working and execution has been shown and each of the screenshot's explanation are briefly done.

REFERENCES

1. Ayman Alserafi, Alberto Abell´o, and Oscar Romero, Toon Calders, Towards Information Profiling: Data Lake Content: Metadata Management, 2016 IEEE 16th International Conference on Data Mining Workshops.
2. Brian Stein, Alan Morrison," The enterprise data lake: Better integration and deeper analytics, Technology Forecast: Rethinking integration", Issue 1, 2014, Retrieved 25, Aug. 2017: www.pwc.com/us/en/technologyforecast/2014/cloud-computing/assets/pdf/pwc-technology-forecast-datalakes.pdf.
3. Chris Campbell, Top five difference between data lakes and data warehouses, JAN 26, 2015, Blue Granite, Retrieved Aug 25, 2017. https://www.bluegranite.com/blog/bid/402596/top-five-differences-between-data-lakes-anddata-warehouses.
4. Chuck Yarbrough, 5 Keys creating killer data lake, retrieved July 21, 2017. http://www.pentaho.com/blog/5-keys-creating-killer-data-lake.
5. Dan Wood, Big data requires a big new architecture, Forbes, Retrieved Aug 8, 2017. https://www.forbes.com/sites/ciocentral/2011/07/21/big-data-requires-abig-new-architecture/#66609cb61157.
6. Gartner.Inc, Gartner Says Beware of the Data Lake Fallacy, STAMFORD, Conn., July 28, 2014, Retrieved 29 Aug, 2017. http://www.gartner.com/newsroom/id/2809117.
7. Hassan Alrehamy Coral Walker, Personal Data Lake With Data Gravity Pull, 2015 IEEE Fifth International Conference on Big Data and Cloud Computing, 26-28 Aug. 2015, Dalian, China.
8. Huang Fang, Managing Data Lakes in Big Data Era: What's a data lake and why has it became popular in data management ecosystem, The 5th Annual IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems, June 8-12, 2015, Shenyang, China. [9] James Dixon, Pentaho, Hadoop and Data Lakes, Retrieved 10 Aug 2017. https://jamesdixon.wordpress.com/2010/10/14/pentaho-hadoop-and-datalakes/
9. Jeffrey Dean, Sanjay Ghemawat, MapReduce: Simplified Processing on large cluster, Communication of the ACM, Vol. 51, No. 1, Jan 2008.
10. Boley, D.; Gini, M.; Gross, R.; Han, E.; Hastings, K.; Karypis, G.; Kumar, V.; Mobasher, B.; and Moore, J. 1999. Partitioning-based clustering for web document categorization. Decision Support Systems 27:329-341.
11. Craven, M.; DiPasquo, D.; Freitag, D.; McCallum, A.; Mitchell, T.; Nigam, K.; and Slattery, S. 1998. Learning to extract symbolic knowledge from the World Wide Web. In AAAI98, 509-516.
12. Dhillon, I. S., and Modha, D. S. 1999. Concept decompositions for large sparse text data using clustering. Technical Report RJ 10147, IBM Almaden Research Center. To appear in Machine Learning.
13. Duda, R. O., and Hart, P. E. 1973. Pattern Classification and Scene Analysis. New York: Wiley.
14. Frakes, W. 1992. Stemming algorithms. In FYakes, W., and Baeza-Yates, R., eds., Information Retrieval: Data Structures and Algorithms. New Jersey: Prentice Hall. 131-160.
15. Hartigan, J. A. 1975. Clustering Algorithms. New York: Wiley.
16. Makhoul, John; Kubala, Francis; Schwartz, Richard; and Weischedel, Ralph (1999); Performance measures for information extraction, in Proceedings of DARPA Broadcast News Workshop, Herndon, VA, February 1999
17. Perry, James W.; Kent, Allen; Berry, Madeline M. (1955). "Machine literature searching X. Machine language; factors underlying its design and development". American Documentation. **6** (4): 242. *Doi*: *10.1002/asi.5090060411*.

18. van Rijsbergen, Cornelis Joost "Keith" (1979); Information Retrieval, London, GB; Boston, MA: Butterworth, 2nd Edition, ISBN 0-408-70929-4
19. Baeza-Yates, Ricardo; Ribeiro-Neto, Berthier (1999). Modern Information Retrieval. New York, NY: ACM Press, Addison-Wesley, Seiten 75 ff. ISBN 0-201-39829-X

AUTHORS PROFILE

First Author: Kimmi Kumari I am a pursuing PhD under Dr. M Mrunalini and my research Centre is M S Ramaiah Institute of Technology, Bangalore. My area of research is big data with security. I have completed my masters of computer applications in the year 2014 and have published two papers in an international conferences and journal till date. My both the papers were purely survey papers in the area of big data processing: issues, techniques and challenges. I have registered for PhD in the year 2017 and have completed my BCA from BIT Mesra, Ranchi in the year 2011. Email: - kimmi.msrit@gmail.com

Second Author: Dr. M MRUNALINI
Dr. M. Mrunalini is working as Assistant Professor in the Department of Master of Computer Applications. Her areas of interests are data warehouse & ETL, software security, big data, software engineering, and software performance engineering.
Email: - mrunalini@msrit.edu