International Journal of Engineering & Technology, 8 (1.11) (2019) 40-46

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Developing a System for Big Data Discovery

Sakhr Saleh, Fathy Eassa, Kamal Jambi

Computer science Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia
**E-mail: Sakhrsalh966@gmail.com*

Abstract

In this research, we developed a system for collecting metadata of existing Big Data in an enterprise or organization. All collected metadata are stored in metadata storage. The collected Meta Data help in managing the existing Big Data. Also, in this research, we developed a technique for discovering simple knowledge from existing Big Data (Twitter & websites) and extracted the correlation between different Big Data. Since Big Data are distributed across a large number of remote machines, we used a mobile agent technology to build our system for reducing the discovery time. The mobile agent migrate to the remote machine for discovering the required data and in consequence reduce the transportation time.

Keywords: *Big Data; Metadata; Knowledge Discovery; Mobile agent.*

1. Introduction

When we work with Big Data (BD), we face many challenges and issues, such as management and transportation challenges, as BD are distributed across a large number of remote machines and stored in large, rapidly increasing volume. The first article to use the term Big Data [1] in October 1997 at ACM was by Cox and Ellsworth, entitled "Managing Big Data for Scientific Visualization" They stated that "Visualization provides an interesting challenge for computer systems: data sets are generally quite large, taxing the capacities of main memory, local disk, and even remote disk. We call this the problem of BD. When data sets do not fit in the main memory (in the core), or even on the local disk, the most common solution is to acquire more resources" [2].
BD Characteristic was presented by a 3V model. Then the model has been extended to five types. The characteristics of such BD will be modeled using HACE (Heterogeneous, Autonomous, Complex, Evolving) Theorem. HACE is the theorem that models BD characteristics which start with massive volume, non-homogeneous, autonomous sources with distributed control and seeks to explore the complex and evolving relationship between data. The 5V models explain as the following [3, 4]. Volume is considered where data size is shifting from Terabytes to Zettabytes. The high-volume of data being generated comes in heterogeneous formats and multiple sources (Variety). Besides, the data has no standard schema as digital asset storage shifted from structured style data storage to contain semi-structured and unstructured data.
Also, the rate (Velocity) at which the data is being generated is very fast which has shifted our focus from data sets (batch) to streaming data. Value is associated with understanding the cost of the BD storage. Moreover, Veracity represent the era of data pollution that needs cleansing. Therefore, it is essential to definethe accuracy of the data by eliminating the noise and make to assure data quality and decisions that are made from the collected data are accurate and efficient.
The following section covers some related work. The architecture of the system is presented in section 3. This is followed by a section on implantation and testing. At the end there is a conclusion.

2. Related work

In this section, we demonstrate the related work; we group them by type as social media and free text. Free text is covered in [5] where the author proposes five components: Metadata Discovery, Metadata collection, Metadata Governance, Metadata storage and Metadata distribution to be managed enterprise BD metadata, because BD introduce large volumes and different data formats. The GLEAN system provides a data discovery environment for users of large scientific computing. GLEAN uses three types of metadata: fine -grained metadata, provenance metadata, and a summary of the datasets. The authors have used the Granules cloud runtime to orchestrate the MapReduce computations that extract metadata from the datasets. There several components, however “Customizable Datasets” component allows users to select or customize the dataset and they can download specific portions of the dataset that will be injected into their computation [6].
Oracle shows a combination of the Oracle BD Appliance and Digital Reasoning Synthesys software. This softwarte has the ability to analyze many millions of documents in a matter of hours. Synthesys is a software platform for automatically making sense of BD. Moreover, it is integrated with Cloudera's Distribution, including Apache Hadoop (CDH). In fact, Synthesys makes use of algorithms and machine learning methods in a three-phase process called "Read-Resolve-Reason". It analyzes scale horizontally to virtually massive size of corpus. Synthesys uses a combination of model-building and unsupervised learning, and discovers the information as a human would

– in context and without the need for a pre-defined ontology, it understands related terms and associations to improve entity recognition, and a contextual understanding of concepts across large sets of text. Extracted information will be stored in a knowledge graph to process data continuously. This will be followed by and a deeply refinement processes [7].

There are many resources for BD. Email is one of them, and authors of [8] claim that usage of e-mail becomes very huge which make it the most used knowledge tool. That paper shows a tool that has been named EKE (Email Knowledge Extraction) which digs into information that could be found in emails of employees. Areas of interest are detected automatically by EKE for picking out key phrases from email messages. EKE is designed in such a way that it is closely integrated into email of client and fits well with the work they do in a natural manner.

Regarding the social media world, authors of [9] present an SABESS project that suggest a combined structural and content based analysis approach. They use social network analysis to identify tweets which are reliable and use some content analysis techniques to summarize some key facts. The ActiveMQ is used as a core massaging system. Then; the crawler stores the tweet in the repository in which they are fed in the messaging system. Moreover, tweets are formatted in JSON format and be coded with various user and tweet related metadata. Some tools are used to do some analysis where tweets' content is parsed and enhanced with additional metadata descriptions before being transferred to the outgoing queue. With the help of content analysis, additional facts about the emergency are obtained from the Tweet text, User data from Twitter, tweet metadata, credibility information from the social network analysis and emergency information. In other words, information obtained from the tweet content are used to construct emergency summaries through a matchmaking process.
Also, a framework for managing BD is presented by authors of [10]. The propose work is done on cloud distributed systems, and introduce an agent-based architecture for metadata extractor of BD. The architecture consists of many mobile as well as stationary agents. The required knowledge is brought by mobile agents that migrate to remote machines that include BD and given back to the main server. The BD manager consists of two sub-managers. They are metadata extraction sub-manager and knowledge discovery sub-manager. The metadata extraction sub-manager extracts and retrieves metadata of the BD on the cloud machines and stores them in metadata storage. Then the knowledge discovery sub-manager discovers the required knowledge or information that is required by the user.

Another work is presented in [11]. It provides an overview of unstructured data, challenges, technology, and data manager implementation. They describe a systematic way of flowing of the unstructured data, collected data, stored data. This paper introduces an unstructured data framework for managing and discovering using the 3Vs of BD. They are Variety, Velocity, and Volume. It includd service-based, metadata storage and data preparation. The development processes in this paper is implemented in Python, build up lexicon and calculated sentiment score.

In [10], the authors proposed an approach for recommending hash tags for tweets. This means providing easy indexing and the manageable search of tweets. Authors of [11] developed a binary language classifier for tweets based on the Naive Bayes method to discriminate between English and non-English language tweets. Then they applied a Latent Dirichlet Allocation (LDA) model in the context of tweet hash tag recommendations. They select keywords as hash tag recommendations. This determines the topic distribution of a tweet which follows by counting the number of words in the top five topics to determine the top words for every topic. At the end, the final result is going to be a set of keywords that is the general topic of a tweet. However, in [11], authors apply LDA to topic model tweets and use the Machine Learning for Language Toolkit (MALLET) API as the implementation of LDA in a Java environment to suggest a relevant topic being discussed in that tweet.

3. System architecture

Fig. 1 shows the System architecture of our system and the following is an elaboration on the components of the system. The architecture shows many elements for collecting metadata and discovering knowledge.

3.1. Big Data Storage

This block represents the data source for our system, which are Twitter and the contents of web pages. These data sets will be explained in the system design section.

3.2. Metadata Collector

We collect metadata for Twitter massages and web page contents by Metadata collector block. We have two types of metadata collector. The first is the Twitter metadata collector and the second is the free text metadata collector. We are using Twitter APIs to collect Twitter metadata and extract web pages metadata by using Google WebCrawler. These metadata will be stored in metadata storage, which will be discussed in the tweets and free texts metadata section.

3.3. Topic Modeler

The core of our system is a topic modeler, which trains topics from the input corpus of documents, and infers topics for new documents. In this block, we used the MALLET tool kit as an implementation for topic modeling. This tool kit is open source, so we used a number of classes, modified it and aggregated it into our system. Topic Modeler takes its input from Metadata collector, which collects tweets or web pages metadata and extracts a message body (tweet) or a text (from the page content) and then sends it as input to the topic modeler.

3.4. Metadata Storage

We store tweets, free text metadata and their topics in a relational database, which is used by Topic & Metadata Retrieval to access topics and metadata for end user queries.

3.5. Metadata Retrieval

When the end user issues queries via the user interface, our system will find the topics for these queries by the topic modeler and send a request to Metadata Retrieval to retrieve all tweets or free text metadata on the same topics.

3.6. Querying Unit

This block is a user interface; it allows the end user to interact with our system.
G. Knowledge Discovery
Once we have obtained metadata for tweets or free text, the Knowledge Discovery block will extract some knowledge from the BD storage using mobile agents

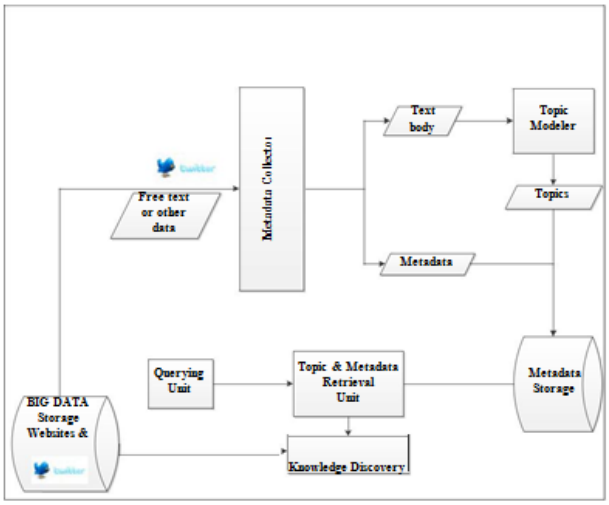


Fig. 1: System architecture

4. Implementation and testing

In this section, we will demonstrate the most important methods in our system and explain the GUI of the system. We also present the tools and libraries that are used to implement the system. Finally, we will test the scenario and explain the results.

4.1. Implementation

Our system is completely implemented in Java; we built a lot of classes and methods to implement it. In addition, we modified several open source tools and libraries to ensure their integration with our system and then embedded them in the system. In this section, we will show the most important methods, tools and libraries that we are using in the system.
1) *Tools and Libraries:* In the implementation phase, we used, modified and embedded several open source tools and libraries in the system, to collect and classify tweets and websites' contents.

2) *MALLET (MAchine Learning for LanguagE Toolkit):* MALLET is open source software, and it provides a Java-based package for statistical natural language processing, topic modeling, document classification and clustering. We used MALLET as an implementation for topic modeling, which uses the LDA algorithm to train topics.
3) *Twitter4j:* Twitter4j is a Java library for Twitter APIs which is open source software. We used it to collect public tweets from Twitter by using search and get/statuses/user_timeline and other REST APIs.
4) *JADE:* JADE is middleware that facilitates the development of multi-agent systems and it is an Open Source library. It includes a runtime environment where agents can live, libraries to build the agents and it also has administration and monitoring tools.
5) *Crawler4J:* Crawler4J is a Java Open Source library that provides Java packages to collect data from websites.

4.2. Testing and results

In this section, we show the GUI of our application and applya testing scenario on the application, then discuss the results.
1) *Application GUI:* There are six parts to the GUI, which are the training, testing, data collector, discover knowledge, user's request and results parts.

In the training part, we can write the path of the training dataset and the path of the features vectors, which is provided by the import dataset process. Also, we can specify the number of topics and the path of the training model or inference file. This training model is generated by the train topics process. Tweets and websites metadata are collected by metadata collectors which are presented in the data collector part TMC (Twitter Metadata Collector) and FMC (Free Text metadata Collector). In the test and request parts, the user can write his request then click on the find topics button to find related topics. The results will appear in the results part which are tweets and websites' documents related to the user's request. Then, using the knowledge discovery part, the user can present charts to visualize the retrieved data.

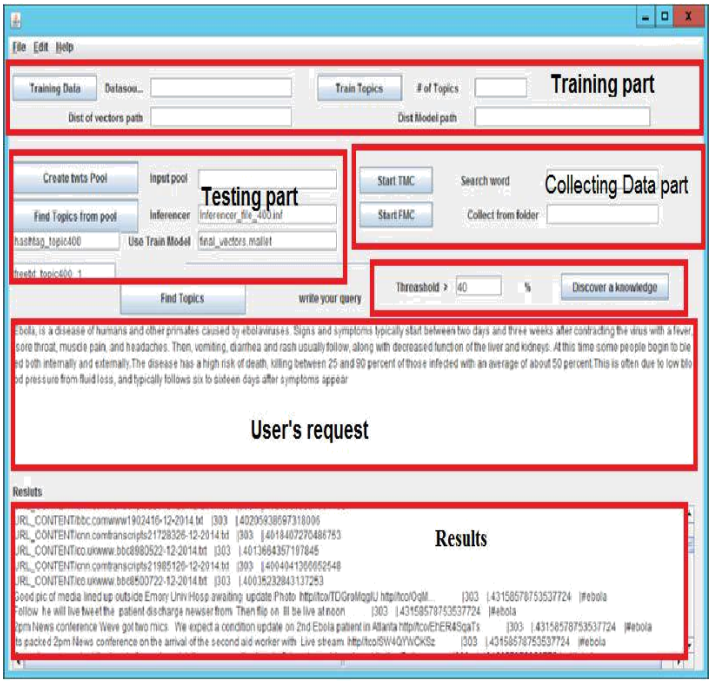


Fig. 2: System GUI

2) Testing Scenario: In this scenario, we used a training model that trains 400 topics from the training corpus, and we want to find related tweets and websites that talk about Ebola so, for this scenario, we put a text that describes Ebola and its signs into the User's request like Fig. 2, then press find related topics to see the results. This is a request or a text for which we want to retrieve related tweets and websites. "Ebola is a disease of humans and other primates caused by ebolaviruses. Signs and symptoms typically start between two days and three weeks after contracting the virus with a fever, sore throat, muscle pain, and headaches. Then, vomiting, diarrhea and rash usually follow, along with decreased function of the liver and kidneys. At this time some people begin to bleed both internally and externally. The disease has a high risk of death, killing between 25 and 90 percent of those infected with an average of about 50 percent. This is often due to low blood pressure from fluid loss, and typically follows six to sixteen days after symptoms appear" [12]. The system retrieves only tweets and websites' documents that have a related probability greater than the threshold. In this scenario, we set the threshold as greater than 40%. There are two mobile agents that start working when a user runs his request, which are the tweets mobile agent and the free text mobile agent. These mobile agents have two tasks. The first one is retrieving related data from the destination machines, and the second is extracting knowledge from these data. Fig. 3 shows the JADE Remote Management Agent (RMA) which handles the GUI interface and shows all of the participating agents and containers.

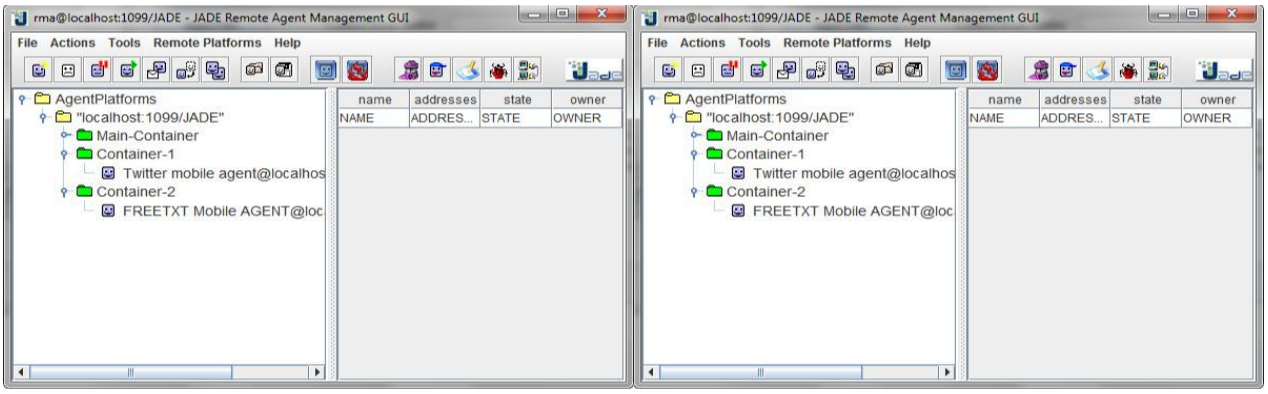


Fig. 3: JADE Remote Management Agent (RMA)

5. Results

In this section, we will show sample results for the testing scenario in section 4.3. These results consist of retrieved data and charts that help us to visualize the result and extract knowledge from it.

1) Retrieved data: Table 1 and 2 show the results of the related tweets and websites that are related to the user's request about Ebola.
2) Knowledge discovery: Our application creates charts to visualize the data; these charts will help us to extract simple knowledge from the results. We will explain these charts in detail and present a value of these charts. Fig. 4 shows the total tweets per year for the Ebola trend. In this chart, we have more than 1000 tweets in 2014 for the Ebola trend, so we can say, for example, 2014 is an Ebola year.

Table 1: documents and urls related to the ebola request

Documents	Topic	Weight	URLs
cnn.comus11565823-12-2014.txt	303	0.82	http://us.cnn.com/2014/04/11/health/ebola-fast facts/index.html?hpt=wo_r1

bbc.comwww11990223-12-2014.txt	303	0.69	http://www.bbc.com/news/world-us-canada-29628622
co.ukwww.bbc8960022-12-2014.txt	303	0.67	http://www.bbc.co.uk/news/world-africa-26835233
cnn.comtranscripts21996426-12-2014.txt	303	0.4	http://www.nejm.org/doi/full/10.1056/NEJMoa1411100

Table 2: tweets related to the ebola request

Tweets	Topic	Hashtag weight	Hashtag name
Ebola virus scans slowing down the process at passport control Expect delays Health Minister Aaron Motsoaledi says Cabinet decided on total ban on traveling to countries affected by Ebola WATCH US flight held as passenger jokes 'I have Ebola' http//tco/H8tTjcZ4s0 http//tco/2xNF7Swlh6	303	43.0	#ebola
My thoughts and prayers to the family of Nigerian Nurse Who Treated Ebola Patients http//tco/O9hPN70pfI via			
Plane on lockdown in Dublin Airport after man claims to have he does not have Ebola http//tco/wVXnNtK7CY http//tco/VRW8RWbqnw			
Ebola Fact A person infected with is not contagious until symptoms appear http//tco/1zZJaP6HSa http//tco/SB…			

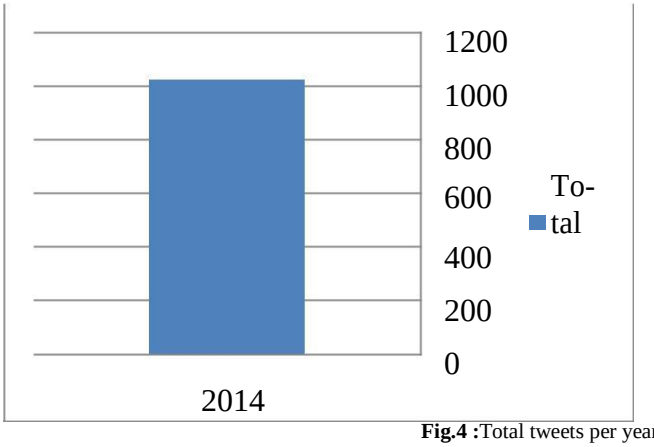
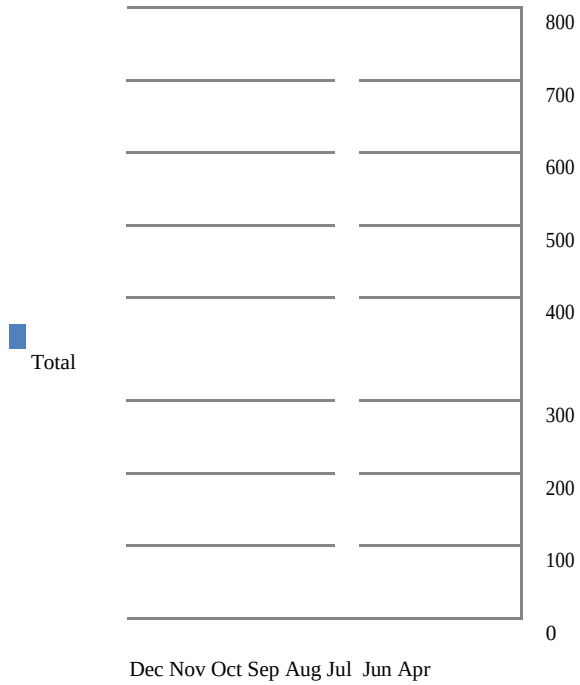


Fig.4 : Total tweets per year for the Ebola trend

The next chart is more detailed and it shows the total tweets per month for the Ebola trend. In this chart, we note that the discussion about Ebola does not start at the begin ning of 2014 but in April, and the maximum value of this trend is in August, with more than 700 tweets, so we can obtain more details and create a chart for only August to discover more knowledge.



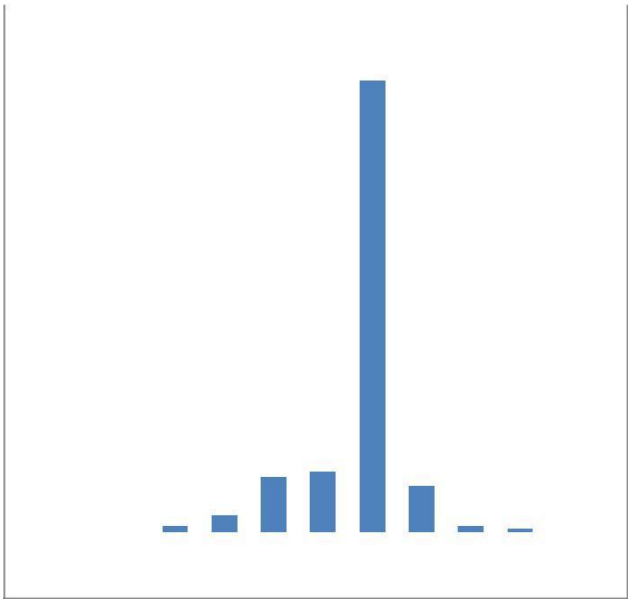


Fig. 5: Total tweets per month for Ebola trend

The total tweets for August per day in chart 6 gives us more details about Ebola. Clearly, from this chart, we note that the tweets curve increases on some days and decreases on others and it reaches a maximum every 7 days. The first day of August in 2014 is a Friday, so we can extract important information from this chart, which says; users' interest in the Ebola trends mid-week; in other words, this trend is less important at weekends.

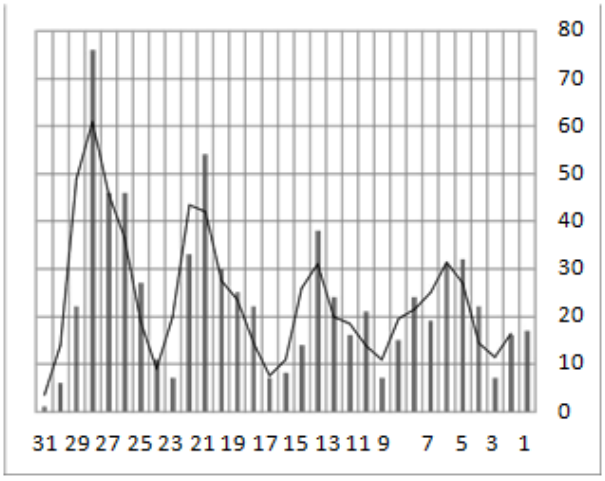


Fig. 6: Total tweets for Aug per day

Lastly, chart 7 shows the total tweets for 2014 per day of the week. This chart gives us an overview of the Ebola trend and proves that the users are active with regard to this trend mid-week, then we can make a decision. We can add programs, topics related to same trend on Tuesday, Wednesday and Thursday to discuss it and introduce solutions.

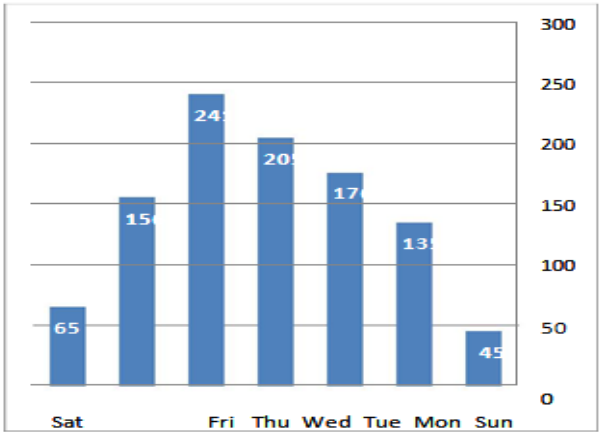


Fig. 7: Total tweets for 2014 per day of week

5. Conclusion and future work

In this research paper, we built a technique and manager for collecting metadata of existing BD in an enterprise or organization. All collected metadata are stored in metadata
storage. We built a technique to discover simple knowledge from the existing BD (Twitter & websites) and extracted the correlations between different BD by using the topic ID.
Since BD are distributed across a large number of remote machines, we use mobile agents technology to build our managers. We used the MALLET (Machine learning for language toolkit) tool kit (open source) as an implementation for topic modeling, which uses the LDA algorithm to train topics. We used mobile agents and metadata of BD to solve the BD transportation challenge in addition to the management challenge.
In future work, we will focus on improving the overall system performance and database scalability by building our system on a Hadoop cluster and creating metadata storage on an NoSQL database. Another concern is that we want to build a real time data collector that uses Twitter stream APIs to collect data from Twitter. This approach allows us to extract the topics and refresh the metadata storage in real time.

References

[1] Press, "A Very Short History of Big Data," [Online]. Available: http://www.forbes.com/sites/gilpress/2013/05/09/a-very-short-history-of-big-data
[2] M. Cox and D. Ellsworth, "Managing Big Data for Scientific Visualization," ACM, October 1997
[3] H. Umar, F. Eassa, K. Jambi and M. Abulkhair, “Big Data Knowledge Mining”, International Journal of Advanced Computer Science and Applications 7(11), November 2016.
[4] H. Umar, Big Data Knowledge Mining, Thesis for a Master Degree of Computer Science, FCIT, KAU, Jeddah, Saudi Arabia, Dec. 2016.
[5] G. Vemuganti, "Metadata Management in Big Data," Infosys labs Briefings, VOL 11 NO 1, 2013
[6] S. L. Pallickara, S. Pallickara, M. Zupanski and S. Sullivan, "Efficient Metadata Generation to Enable Interactive Data Discovery over Large-scale Scientific Data Collections," 2nd IEEE International Conference on Cloud Computing Technology and Science, 2009.
[7] Oracle, "Big Data and Natural Language: Extracting Insight From Text," September 2012. [Online]. Available: http://www.oracle.com/ocom/groups/public/@otn/documents/webconte nt/1863164.pdf. [Accessed 22 04 2015].
[8] S. Tedmori, T. W. Jackson and D. Bouchlagem, "Optimising the Email Knowledge Extraction System to Support Knowledge Work," ECIS, pp. 681-691. University of St. Gallen, 2007.
[9] B. Klein, X. Laiseca, D. Casado-Mansilla, D. Lopez-de-Ipina and A. P. Nespral, "Detection and Extracting of Emergency Knowledge from Twitter Streams," UCAmI 2012, LNCS 7656, p. 462–469, 2012.
[10] F. E. Eassa, H. Al-Barhamtoshy, A. Almenbri, O. H. Younis and K. Jambi, "An Architecture for Metadata Extractor of Big Data in Cloud Systems," International Journal of Scientific & Engineering Research, Volume 5, Issue 1, January 2014.
[11] H. M. Al-Barhamtoshy and F. E. Eassa, "A Data Analytic Framework for Unstructured Text," Life Science Journal, pp. 339-350, 2014;11(10).
[12] Karandikar, Clustering short status messages: A topic model based approach, University of Maryland, 2010.