International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2015): 6.391

Implementing HACE Theorem for Big Data Processing

Prema Gadling[1], Mahip Bartere[2]

[1]Student of Master of Engineering (Computer Science & Engineering), Raisoni College of Engineering and Management, Amravati, India

[2] Assistances Professor, Computer Science and Engineering, Raisoni College of Engineering and Management, Amravati, India

Abstract: ***The aim is propose to elaborate a HACE theorem that states the characteristics of the Big Data revolution, and proposes a Big Data processing model from the data mining view. Here, Data comes from everywhere like sensors, media sites and social media etc. In this useful data can be extracted from this big data using data mining technique for discovering interesting patterns. As enhancement we propose Detection of emerging topics from social networks of big data. Specifically, we focus on mentions of user links between users that are generated dynamically (intentionally or unintentionally) through replies, mentions, and retweets. In this paper, we are going to talk how effectively analysis is done on the data which is collected from the Twitter using Flume. Twitter is an online web application which contains rich amount of data that can be a structured, semi-structured and un-structured data. We can collect the data from the twitter by using big data eco-system using online streaming tool Flume. And doing analysis on Twitter is also difficult due to language that is used for comments. And, coming to analysis there are different types of analysis that can be done on the collected data. And this paper provides a way of analysis data using hadoop which will process the huge amount of data on a hadoop cluster faster in real time***.

Keywords: Big Data, data mining, REST API, HACE THEOREM, HDFS, LDA, Sentiment analysis.

1. Introduction

1.1 Data mining

Data mining is the technology to extract the knowledge from the data. The data to be mined varies from a small data set to a large data set i.e. big data. The data Mining environment produces a large volume of the data. The information retrieved in the data Mining step is transformed into the structure that is easily understood by its user.

1.2 Big Data

Big data are the large amount of data being processed by the Data Mining environment. In other words, it is the collection of data sets large and complex that it becomes difficult to process using on hand database management tools or traditional data processing applications, so data mining tools were used. Big Data are about turning unstructured, invaluable, imperfect, complex data into usable information.

1.3 HACE Theorem

HACE Theorem used to characterizes of the Big Data revolution, and proposes a Big Data processing model, from the data mining perspective. The characteristics of HACE make it an extreme challenge for discovering useful knowledge from the Big Data. The HACE theorem suggests that the key characteristics of the Big Data are

1.3.1 Huge with heterogeneous and diverse data sources

One of the fundamental characteristics of the Big Data is the huge volume of data represented by heterogeneous and diverse dimensionalities. This huge volume of data comes from various sites like Twitter, MySpace, Orkut and LinkedIn etc.

1.3.2 Decentralized control

Autonomous is the similar to the characteristic of a distributed computing platform. The resources are spread across the different system. It is stand alone working environment and there is no centralized sever.

1.3.3 Complex data and knowledge associations:-

Complex and evolving data is similar to our facebook friends. As a person's number of friend increase the representation of data get more and more complex. As, the list grow over time it can be term as an evolving data.

1.4 Hadoop

Hadoop is an Apache open source framework written in java that allows distributed processing of large datasets across clusters of computers using simple programming models. Hadoop use HDFS (Hadoop Distributed File System) file system**.** The Hadoop Distributed File System (HDFS) is based on the Google File System (GFS) and provides a distributed file system that is designed to run on large clusters (thousands of computers) of small computer machines in a reliable, fault-tolerant manner. HDFS uses a master/slave architecture where master consists of a single NameNode that manages the file system metadata and one or more slave DataNodes that store the actual data. Benefit of using Hadoop is distributed storage , Distributed Processing , Security, Reliability, Speed , Efficiency, Availability, Scalability and lots more. This is the reason of using Hadoop for tweet processing.

The following paper shows the how to process/analysis big data. The paper has been described as follows, in section I Introduction about data mining, big data and HACE Theorem. In section II we discuss about the twitter and how to get twitter data with the help of REST API. In section III

he proposed system flow. In section IV describes proposed system.

2. Twitter and twitter REST API

Twitter is an online social networking service that enables users to send and read short 140-character messages called "tweets". Registered users can read and post tweets, but unregistered users can only read them.

A REST API defines a set of functions which developers can perform requests and receive responses via HTTP protocol such as GET and POST. Twitter provides a REST API which you can query to get the latest tweets, you can provide a search query (or hash tag) and it will return the results in JSON format.

The information can be collected using both forms of Twitter API. Requests to the APIs contain parameters which can include hash tags, keywords, geographic regions, and Twitter user IDs. Responses from Twitter APIs is in JavaScript Object Notation (JSON) format. JSON is a popular format that is widely used as an object notation on the web. Twitter APIs can be accessed only via authenticated requests. Twitter uses Open Authentication and each request must be signed with valid Twitter user credentials. Open Authentication (OAuth) is an open standard for authentication, adopted by Twitter to provide access to protected information. The authentication of API requests on Twitter is carried out using OAuth. Twitter APIs can only be accessed by applications. Below we detail the steps for making an API call from a Twitter application using OAuth:

1) Applications are also known as consumers and all applications are required to register themselves with Twitter4. Through this process the application is issued a consumer key and secret which the application must use to authenticate itself to Twitter.
2) The application uses the consumer key and secret to create a unique Twitter link to which a user is directed for authentication. The user authorizes the application by authenticating himself to Twitter. Twitter verifies the user's identity and issues a OAuth verifier also called a PIN.
3) The user provides this PIN to the application. The application uses the PIN to request an "Access Token" and "Access Secret" unique to the user.
4) Using the "Access Token" and "Access Secret", the application authenticates the user on Twitter and issues API calls on behalf of the user.

The "Access Token" and "Access Secret" for a user do not change and can be cached by the application for future requests. Thus, this process only needs to be performed once, and it can be easily accomplished using the method GetUserAccessKeySecret.

3. Proposed System

For doing twitter data analysis first data is collected using Access token in local HDFS. Tweets are preprocesses for removing noise and meaningless symbols. After that LDA and sentiment can be used for twitter posts analysis.

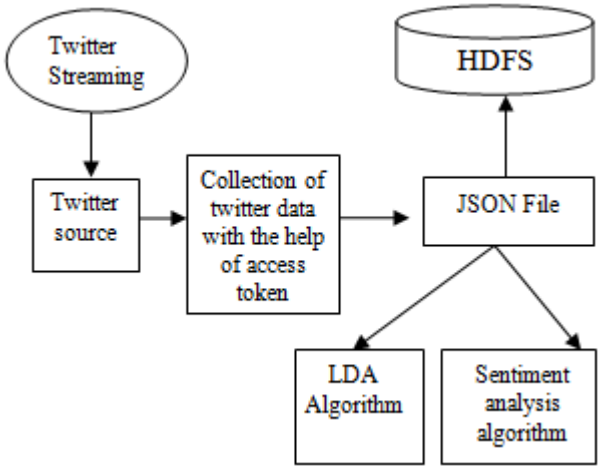

Figure 1: Installation & Work Flow

The above diagram shows the complete step wise working of twitter posts analysis.

3.1 Create Access token [Open Authentication (OAuth)] for collection twitter data

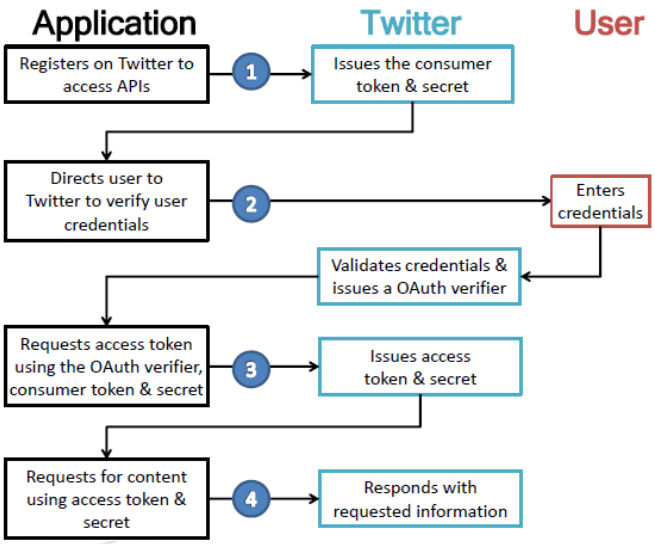

Figure 2: OAuth workflow

3.2 Hadoop

Hadoop is a open source framework that developed by apache software foundation. Hadoop are the most widely used models used today for Big Data processing. Hadoop is an open source large-scale data processing framework that supports distributed processing of large chunks of data using simple programming models. The Apache Hadoop project consists of the HDFS and Hadoop Map Reduce in addition to other modules. The software is modeled to harvest upon the processing power of clustered computing while managing failures at node level. All the modules in Hadoop are designed with a fundamental assumption that hardware failures are common and should be automatically handled by the framework. Apache Hadoop consists of a storage part, known as Hadoop Distributed File System (HDFS), and a

processing part called MapReduce. Apache Hadoop MapReduce and HDFS components were inspired by Google papers on their MapReduce and Google File System Hadoop splits files into large blocks and distributes them across nodes in a cluster. To process data, Hadoop transfers packaged code for nodes to process in parallel based on the data that needs to be processed. This approach takes advantage of data locality nodes manipulating the data they have access to allow the dataset to be processed faster and more efficiently than it would be in a more conventional supercomputer architecture that relies on a parallel file system where computation and data are distributed via high-speed networking.

3.3 Latent Dirichlet allocation (LDA)

LDA is a collection of words. Each topic contains all of the words in the corpus with a probability of the word belonging to that topic. It involves setting up the requisite count variables, randomly initializing them, and then running a loop over the desired number of iterations where on each loop a topic is sampled for each word instance in the corpus. Following the Gibbs iterations, the counts can be used to compute the latent distributions _d and _k.

The only required count variables include nd;k, the number of words assigned to topic k in document d; and nk;w, the number of times word w is assigned to topic k. However, for simplicity and e_ciency, we also keep a running count of nk, the total number of times any word is assigned to topic k. Finally, in addition to the obvious variables such as a representation of the corpus (w), we need an array z which will contain the current topic assignment for each of the N words in the corpus.

Table 1: Latent Dirichlet allocation

Input: words w ∈ documents d
Output: topic assignments z and counts $n_{d,k}, n_{k,w}$, and n_k
begin
  randomly initialize z and increment counters
  foreach *iteration* do
    for $i = 0 \rightarrow N-1$ do
      $word \leftarrow w[i]$
      $topic \leftarrow z[i]$
      $n_{d,topic}$-=1; $n_{word,topic}$-=1; n_{topic}-=1
      for $k = 0 \rightarrow K-1$ do
        $p(z = k|\cdot) = (n_{d,k} + \alpha_k)\frac{n_{k,w}+\beta_w}{n_k+\beta\times W}$
      end
      $topic \leftarrow$ sample from $p(z|\cdot)$
      $z[i] \leftarrow topic$
      $n_{d,topic}$+=1; $n_{word,topic}$+=1; n_{topic}+=1
    end
  end
  return z, $n_{d,k}, n_{k,w}, n_k$
end

3.4 Sentiment analysis algorithm

Sentiment Analysis is to detect the polarity of text in consideration in textual form. It is also known as opinion mining as it derives the opinion of the speaker or the user about some topic. The sentiment analysis algorithm we use in our project is based on a Naive Bayes Classifier. Since Naive Bayes is fast, space efficient, and not sensitive to irrelevant features, in this research we used the Naive Bayes classifier which is based on Bayes' theorem.

$$P(w|T) = \frac{P(w) \cdot P(T|w)}{P(T)}$$

where w is a sentiment word, T is a Twitter message

Bayes's theorem is based on strong independence assumptions. Therefore, the probabilistic model for a classifier can be described as:

$$R = P(positive|T) - P(negative|T)$$
$$= P(positive)P(T|positive) - P(negative)P(T|negative)$$
$$= P(positive)\prod P(T|positive) - P(negative)\prod P(T|negative)$$

Comparing the probabilities P(positive|T) and P(negative|T), the larger probability indicates that the class label value has a higher probability to be actual label. If R is larger than 0, then predict positive attitude is more likely to be true, otherwise, predict negative attitude has more likely to be true.

During the sentiment analysis, the Naive Bayes classifier classifies a Tweet into a positive class or a negative class by comparing the words in each Tweet. Each word will be labeled with "positive" and "negative" coming from the lexicon. In the Naive Bayes classification, the number of sentiment words is counted. If more positive words are used than negative in a Tweet, then the Tweet could be labelled as positive, otherwise if less positive words presented in a Tweet than negative ones, the Tweet could be labelled as negative. A neutral label word is ignored in this study since it contains no valuable information for sentiment analysis.

4. Result

There are several ways to define and analyze the social media data such as Twitter, facebook etc. Here anyone can perform different operations queries in these type of data. But the problem arises when dealing with big data of several types of unstructured data and semi structured data. As twitter post are very important source of opinion on different issues and topics. It can give a keen insight about a topic and can be a good source of analysis. Analysis can help in decision making in various areas. Here in our project we identify the HACE characteristic for twitter big data processing.

This project will give us hands on experience of handling and parallel processing of huge amount of data. Data collection process will introduce us to Java twitter streaming API. We will get exposure to work with prominent parallel

data processing tool: Hadoop. Apache Hadoop is one of the best options for twitter post analysis. We have done analysis on the Twitter data that is stored in HDFS. And then we applied latent Dirichlet allocation (LDA) and sentiment analysis algorithm. Also it do the analysis on real time data, so is more useful. The analysis what I did could be helpful in finding people mood for IPL tweets. And can be helpful in strategy planning. So, here the processing time taken is also very less because Hadoop(Flume, HDFS) are the best methods to process large amount of data in a small time. So it is concluded that processing time, retrieving capabilities and analysis are made very easy when compared to other processing and analyzing techniques for large amounts of data.

References

[1] G. Q. Wu, X. Wu, X. Zhu (January 2014) "Data mining with big data," IEEE Trans. on Knowledge and Data Engineering, vol. 26, no. 1, pp. 97-107.
[2] Sagiroglu, S.; Sinanc, D. (2013) "Big data: A review," Collaboration Technologies and Systems (CTS), International Conference on , vol., no., pp.42,47, 20-24 May 2013
[3] Albert Bifet, Wei Fan, "Mining Big Data: Current Status and Forecast to the Future", SIGKDD Explorations, 14 (2), pp1-5
[4] Mahalakshmi R, Suseela S , "Big-SoSA:Social Sentiment Analysis and Data Visualization on Big Data", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 4, April 2015 , pp 304-306, ISSN : 2278-1021.
[5] Sunil B. Mane , Sunil B. Mane, Yashwant Sawant, Saif Kazi, Vaibhav Shinde , "Real Time Sentiment Analysis of Twitter Data Using Hadoop", (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (3) , 2014, 3098 – 3100 , ISSN:0975-9646.
[6] Ramesh R, Divya G, Divya D, Merin K Kurian , "Big Data Sentiment Analysis using Hadoop ", (IJIRST)International Journal for Innovative Research in Science & Technology,Volume 1 , Issue 11 , April 2015 ISSN : 2349-6010.
[7] https://blog.cloudera.com/blog/ 2012/11/analyz ing-twitter-data-with-hadoop-part-3-querying-semi-structured-data-with-hive/
[8] Michael G. Noll, Applied Research, Big Data, Distributed Systems, Open Source, "Running Hadoop on Ubuntu Linux (Single-Node Cluster)", [online], available at http://www.michael-noll.com/tutorials/running-hadoop-on-ubuntu-linux-single-node-cluster/
[9] "Mining Data from Twitter" from AbhishangaUpadhyay, Luis Mao, Malavika Goda Krishna (PDF)
[10] M. C. Pham, Y. Cao, R. Klamma, et al., "A Clustering Approach for Collaborative Filtering Recommendation Using Social Network Analysis," Journal of Universal Computer Science, vol. 17, no. 4, pp.583-604, April 2011.
[11] Aditya B. Patel, Manashvi Birla, Ushma Nair, "Addressing Big Data Problem Using Hadoop and Map Reduce",6-8 Dec. 2012.
[12] Ankit Darji, Dinesh Waghela," Parallel Power Iteration Clustering for Big Data using MapReduce in Hadoop", International Journal of Advanced Research in Computer Science and Software Engineering,Volume 4, Issue 6, June 2014 ISSN: 2277 128X.

Author Profile

Mahip Bartere pursuing a Ph.D. degree with Master of Engineering (Computer Science & Engineering), from Sipna college of Engineering & Technology, Amravati or BE degree already completed.

Prema Gadling received the Bachelor of Engineering degree from Shree Hanuman Vyayam Prasarak Mandals' College of Engineering and Technology, Amravati. And the currently perusing Master of Engineering (Computer Science & Engineering), Raisoni College of Engineering and Management, Amravati, India