Study of Sentiment Analysis Using Hadoop

Dipty Sharma

Abstract In the current world of Internet people express themselves, present their views and feelings about specific topics or entities using various social media application. These posts from users present a huge opportunity for the organizations to increase their market value by analyzing the posts and using information in decision making. These posts can be studied using various machine learning and lexicon-based approaches for extracting its sentiments. With more and more people moving to internet, huge data is being produced every second and challenge is to store this large data and process it efficiently in real time to infer knowledge from this data. This paper presents different approaches for real-time and scalable ways of performing sentiment analysis using Hadoop in a time efficient manner. Hadoop and its component tools like MapReduce, Mahout, and Hive are being surveyed in different scholar articles for this paper.

Keywords Sentiment analysis · Twitter · Hadoop · MapReduce · Hive · Mahout

1 Introduction

As e-commerce is gaining popularity, more and more e-commerce sites are being launched. These sites are constantly in need to keep its customers happy and to identify ways to overcome competition. One way to achieve customer satisfaction is to be able to present and display the products based on customer’s interests. With sites like Twitter, Facebook etc., users are provided with a platform to post emotions, views, and likings about various topics, people, product, and services. Opinions expressed in social media can be classified to determine the orientation (negative, positive, and neutral) of the posted text. Sentiment strength and intensity
of the post are determined with the aim to identify opinion and emotion of the user about a specific product or service.

Sentiment of textual posts can be analyzed in three different levels:

- **Document Level**

  Analytical study performed for the whole document, determining the polarity (positivity, negativity or neutrality) of the document on the subject as a whole.

- **Sentence Level**

  Analytical study performed for each statement or sentence where each sentence is expressed as positive, negative or neutral.

- **Aspect Level**

  Analytical study performed at fine-granular level. Each aspect or feature of product is analyzed, and polarity is determined for each feature.

  Technically, following are ways to perform sentiment analysis.

1. **Machine learning Approach**

   Sentiment analysis process can be performed using various machine learning algorithms like Naive Bayes. These learning approaches are based on building classifiers from labeled instances of textual posts. They perform well for the domain on which they are trained.

2. **Lexicon-Based Approach**

   These approaches calculate emotional orientation of a document from the semantic orientation of words or phrases in the document. These dictionary-based approaches consist of dictionary of number of words annotated with their polarity, strength, and semantic orientation.

   With increase in access to Internet and more people coming online and using e-commerce. Textual information on internet is increasing every second, and it is a challenge to read and process this vast data set in efficient manner.

   Hadoop provides a framework (Fig. 1) commonly used by academic and industry for Big Data analysis. It allows collection, storage, retrieval, management, and distributed processing of huge data sets using cluster of computers and simple programming models. Each machine in Hadoop cluster has local storage and can perform local computation. Hadoop is highly performance intensive, scalable, and flexible development framework for parallel processing. Hadoop framework offers reliable service availability by detecting and handling failures at application layer itself.

   Audience of this paper are professionals and practitioners who intend to perform sentiment analysis using Hadoop. Section 2 shows how the steps in sentiment analysis process maps with different components of Hadoop. Section 3 describes scholarly articles on sentiment analysis using Hadoop. Section 4 provides detailed
review on various aspects for performing sentiment analysis in Hadoop. Section 5 concludes on using Hadoop for sentiment analysis.

2 Implementation of Sentiment Analysis in Hadoop

Process of sentiment analysis involves collection of textual data from various sources like blogs, reviews, social sites etc., processing this vast volume of data to remove unwanted, undesirable text, analyzing, extracting sentiment of collected data and integrating with applications to help decision making. A simple process of sentiment analysis is shown in Fig. 2 involves following steps (1) Data Collection (2) Data Preprocessing (3) Analysis.

Following section lists one to one mapping of sentiment analysis process steps in Hadoop Framework Fig. 7.

2.1 Data Collection and Storage

HDFS and Hive provide data storage capabilities in Hadoop. Hadoop Distributed File System (HDFS) provides distributed file system that can store data in different
formats. It provides high-throughput access to application data. Data present in logs or files can be read using MapReduce programming and stored in HDFS. MapReduce core component of Hadoop. It is a java-based framework for writing easy applications which process vast amounts of data in parallel. A MapReduce job splits the input data set into independent chunks which are processed by the map tasks in a completely parallel manner Fig. 3. The framework sorts the outputs of the maps, which are then input to the reduce tasks Fig. 4. Both the input and the output of the jobs are stored in a HDFS. The framework takes care of scheduling tasks, monitoring them and re-executes the failed tasks.

Real-time data from Twitter can be streamed into HDFS using Flume. Flume [1] has a simple and flexible architecture efficiently collecting, aggregating, and moving large amounts of log data. Data from Twitter in the form of Json can be partitioned based on posted dates and stored in Hive for processing. Hive is a data warehouse infrastructure that provides data summarization and ad hoc querying using SQL like commands. Large data present in relational database can also be loaded into HDFS, Hive using Sqoop. Sqoop [2] imports and exports data from numerous relational databases. Figures 5 and 6 show the process architecture of Flume and Sqoop.
2.2 Data Preprocessing

Data preprocessing is natural language processing steps required to clean the textual data from all the unwanted text from the collection of data. Data preprocessing involves cleaning the data: removal of stop words, spell check, conversion of unstructured data into structured data (e.g., happiest → happy), conversion of slangs (e.g., luv → love, OMG → oh my god), conversion of hash tags and
emoticons and extracting important features, removing duplicates, and converting
data formats. External Libraries like NLTK python library and OpenNLP java
library can be plugged with MapReduce or User Defined Functions (UDFs) in Hive
for natural language processing. A Hive provides built-in functions for text pre-
processing Table 1.

| **Table 1** | Hive pre-post processing functions |
|-------------|-----------------------------------|
| **Built-in** | **Usage**                          | **Output**          |
| sentences   | Tokenizes a string into words and sentences, returns an array of words | Array               |
| explode     | Returns one row for each element from the array | Rows               |
| get_json_object | Returns json string of the extracted json object | String             |
| json_tuple  | Returns a tuple of values using one function from json string | Tuple              |

2.3 Sentiment Classification

Sentiment analysis of text data is essentially the classification of text, based on the
strength and polarity (positive/negative) of the opinion words that defines the text.
Hadoop provides framework for users to develop their own sentiment analysis algorithms using lexicon dictionary, available APIs or external programs. These algorithms can be implemented using MapReduce or Hive UDFs [3] in Hadoop. Hive also provides built-in text analysis functions Table 2. Hive uses N-gram Probabilistic language Model for predicting next word in a sequence of words. Mahout provides machine learning library consisting of different inbuilt classifiers [4–6] (Fig. 7).

| **Table 2** | Hive built-in classifier |
|-------------|--------------------------|
| **Built-in** | **Usage**                | **Output**           |
| ngrams      | Find important topics using a stop word list, trending topics | k most frequently occurring n grams |
| Context_ngrams | Extract intelligence around certain keywords, pre compute search look ahead | k most frequently occurring n grams |
3 Literature Survey

A Speedy Data Uploading Approach for Twitter Trend and Sentiment Analysis Using Hadoop

Gaurav et al. [7] in their paper demonstrated how open source framework of Hadoop can help organizations with real time, cost effective and secure social analytics. Their study presents a distributed Hadoop system using HDFS, Pig, Hive and Oozie for Twitter trend analysis. Querying Twitter post in RDBMS is difficult because of multiple retweets of same post. It is important to identify who is prominent. Their proposed work combined open source software and hardware phenomenon. Data download speed is increased with the help of network-based application. Latency and network delays are removed. Reliable connection between source and sink is established using application on Twitter side. Their work shows how data available in HDFS is processed with help of Hive and Pig. To coordinate and schedule the processes, they used Oozie. Zookeeper helped maintain the configuration information providing distributed synchronization. Their work shows how Hadoop can help increase organization profits by reducing cost, time delays, and security issues.

Real Time Sentiment Analysis of Twitter Data Using Hadoop

Mane et al. [8] use MapReduce programming framework lexicon dictionary-based sentiment analysis at sentence level. Data is collected in real time using Twitter streaming API and stored in HDFS. Collected data is preprocessed to remove the stop words, convert unstructured text to structured text and convert emoticons into
words using OpenNLP implementation. Lexicon dictionary of sentiment words using SentiWordNet is generated. All possible usage of a word is obtained to determine the overall sentiment of the word and updated in the dictionary. They used numbering approach to assign suitable range for different sentiments. To reduce the search time dictionary is stored local memory. Naive Bayes algorithm is implemented using chained MapReduce jobs to process each tweet and assign sentiment to each word. PMR-IR algorithm is implemented to determine orientation of phrases. Sentiment of tweet is computed by aggregating the sentiments for each word in tweet. Overall accuracy of 72% is achieved. Time efficiency is achieved for large data set using Hadoop.

The Evaluation of the Public Opinion a Case Study: MERS-CoV Infection Virus in KSA

Zarrad et al. [9] based their study on methods similar to [10]. Their work is to detect sentiments on Arabic Tweets on MERS Virus using multiple Hive UDFs (User defined functions) using Arabic lexicon. Keyword-based search performed on certain words like MERS-CoV virus is done using Twitter Rest API. Tweets are collected and stored into HDFS using Flume. Because of the complexities of Arabic language, a custom text preprocessing module is developed to clean the text. Sentiment analysis module is developed using Hive UDF. Lexicon based on MPQA is manually updated with 1100 negative and 850 positive words. Lexicon is translated into Arabic for this work. Their proposed sentiment detection algorithm can search up to 5 consecutive composite words in polarity lexicon and can detect negation on opinion. Proposed algorithm can identify positive word affected by non-consecutive negative word. To get the satisfaction measure of efforts undertaken by Heath Ministry on controlling MERS-CoV virus, they collected tweets for four months and evaluated public opinion.

Big Data Sentiment Analysis Using Hadoop

Ramesh et al. [11] proposed sentence level sentiment analysis using lexicon dictionary-based approach. The study shows how accuracy of sentiment detection can be achieved while focusing on speedy processing on vast data sets using Hadoop. They created dictionary of sentiment words with strength (strong, weak) and polarity (positive, negative, negation, and blind negation) for each word. Dictionary consists of all forms of words to avoid stemming each word to increase processing time. Negation and blind negation words that reverse the polarity of sentence are included in the dictionary. Sentiment analysis is performed by combining the polarity of the word its strength to compute the sentiment score with score 0 as neutral polarity. Sentence polarity is aggregation of words score. Polarity of sentence is reversed when blind negation word is identified in the sentence. Their study shows how lexicon-based sentiment analysis implemented in Hadoop performs better than machine learning approaches for the real-time data that is not domain specific.
Cloud-Based Predictive Analytics

Kalvdiya et al. [12] studies are based on machine learning based text classification on cloud environment. They suggested category-based text classification of e-mails, to automatically classify incoming news feeds into appropriate categories. The classification results contained the index of the category label and best associated score output.

For classification training data, they collected set of e-mails from Mahout and Hive user list to train Naïve Bayesian algorithm. E-mails were stored in HDFS, and Mahout Command was used to train the classifier model. Data for training is represented in word-vector format using TF-IDF (term frequency–inverse document frequency), which involves counting word frequencies over large corpus of documents. Feature document vector was created with TF-IDF weighting implemented by Mahout to give more weight to specific topic words. Naïve Bayes classifier model was performance tested by running Mahout testnb command.

To use the classifier model for new text files java program was written using Mahout Libraries. Files in directories were converted into sequential format, and TF-IDF sparse vectors were generated and then classified using the model. Model was run on the cloud using Maven to manage dependencies. During their work they identified that Mahout scales well with massive data sets and performs comparable to other machine learning algorithms.

Research on Public Opinion Based on Big Data

Shang et al. [13] demonstrate how Mahout algorithms can be efficiently used for processing large scale and complex Big Data. Mahout text mining algorithms are presented to handle high-dimensional data. Preprocessing of data was done using TF-Gini algorithm, and processing of text is performed using Mahout algorithms. Their research demonstrated clustering algorithms: canopy and k-means algorithms. Canopy algorithm differs from traditional algorithm as it uses two computing distance methods and computes only overlapping data vectors. Mahout’s built-in Naïve Bayes algorithm for classification, Apriori and FP-tree algorithms for pattern mining are presented. Mahout used Parallel Frequent Pattern Mining. Firstly identifying one-dimensional frequent items and then dividing original data set into different groups based on frequent items. FP-tree is computed for each group. FP-tree is mining to get frequent items, and result is obtained by merging each FP-tree frequent items. The proposed system using parallel computing function of Hadoop Mahout.

Scalable Sentiment Classification for Big Data Analysis Using Naive Bayes Classifier

Liu et al. [14] present scalable implementation of Naïve Bayes algorithm to analyze sentiment sentences of millions of movie reviews. They build Big Data analyzing system using simple MapReduce analyzing jobs and work flow controller (WFC), user terminal, result collector and data parser.
Movie reviews collected was preprocessed through data parser in desired data format. Unwanted context such as punctuation, special symbols, and numbers was deleted and each review was split into one line in data set tagged with sentiment and document id. Data parser would return the desired number of positive and negative reviews on request to WFC. WFC manages work flow of the whole system and stores data set in HDFS for training. Training job, combining job, and classifier job were executed in sequence. The training job builds a model that computes the polarity of each word based on the frequency of word. The combining job generates an intermediate table by associating the test data with the model, excluding the words that appear in test data but not in the model. Classifier job then classifies reviews into positive or negative and output the results into HDFS. The statistics of true positive, true negative, false positive, and false negative were recorded. Result collector would retrieve the model, intermediate table, classification result, and statistics of test from HDFS. User terminal was used to submit user jobs and results were accessible in user terminal after result collected finished collection. The scalability of the algorithm in Hadoop was tested with accuracy of 82% with changing the data set size from one thousand to one million in each class.

Scaling Archived Social Media Data Analysis Using a Hadoop Cloud

Conejero et al. [15] present COSMOS platform for Twitter data analysis on socially significant events. Cardiff Online Social Media Observatory (COSMOS) platform provides mechanisms to capture, analyze and visualize results of real-time data. They demonstrated how this system can scale using OpenNebula Cloud environment with MapReduce using Hadoop for Big Data. COSMOS ingest and archives the spritzer stream using Twitter Streaming API. Approximately, 3.5 million messages per day can be processed using COSMOS API. Collection of virtual machines makes Hadoop cluster. OpenNebula helps decide size and characteristics of Hadoop cluster. Kernel-based Virtual Machine (KVM) manages virtualization within resources. MapReduce paradigm processes each tweet in parallel using SentiStrength. SentiStrength estimates the strength of word in scale of $-1$ to $-5$ for negative opinion and $1$–$5$ for positive opinion. They demonstrated the feasibility of MapReduce using Hadoop for COSMOS and performance benefits achieved by using multiple virtual worker nodes. Greater the number of virtual nodes better the performance.

DOM: A Big Data Analytics Framework for Mining Thai Public Opinions

Prom-on et al. [16] present DOM (Data and Opinion Mining) a mobile data analytics engine for mining Thai public opinions. They present keyword-based sentiment analysis using MapReduce. Messages collected from social networks, blogs, and forums are stored in MongoDB. Twitter data is collected using Search API, and Facebook data is collected using Graph API. Each message is processed using natural processing techniques using MapReduce technique on Hadoop. MapReduce accelerated the analysis speed. Lexicon-based algorithm is developed to measure sentiment of score of each word. They defined five corpora which includes positive,
negative, modifiers, conjunction, and point of interest words with sentiment rating ranging from −5 to 5. LexTo tool is used to tokenize words in each sentence. DOM implements MapReduce program to generate jobs that detects words in each sentence in parallel. Non-related sentences were discarded by matching words in point of interests. DOM then computes the sentiment score using positive and negative words in corpora. Sentiment score is inverted when modifier is adjacent to sentiment words. DOM engine can classify messages and perform sentiment analysis with accuracy of over 75% when compared to Human analysis. DOM engine was tested on general public opinion expressed in social network to determine political climate around end of 2013.

Customer Preference Analysis Based on SNS Data

Kim et al. [10] proposed feature based sentiment detection of social network sites using Hadoop. Twitter data is collected using TwitterAPI, Twitter4J stored and analyzed in multi-dimensional fashion to identify factors that affect customer preferences on smart phones. Their approach used Hannanum Java-based morphological analyzer to process the data into sentiments. Morphological analyzer consists of text preprocessing, morphological analysis, and POS tagging. Synonyms and acronyms in Twitter are collected and processed. They performed multi-dimensional analysis to find the sentiment of each attribute of mobile. Each Twitter table has 4-dimensional tables: mobile and its attributes, sentiment words, mobile carrier, brand, and maker. Analysis on each of this dimension table is done using implementation in Hive and R. Three aspects of Big Data volume, variety, and velocity were handled by making real-time feed from Twitter and analyzing in multi-dimensional fashion to address variety and volume in Hadoop.

The Impact of Cluster Characteristics on HiveQL Query Optimization

Joldzic et al. [17] analyzed the impact of cluster characteristics on HiveQL query optimization. Non-relational databases were developed to improve processing of Big Data. In this paper, they discussed the transfer of data from relational databases MySQL to distributed data storage HDFS using Apache Sqoop. MapReduce is used by Sqoop to import and export data using parallel operations in fault-tolerant manner. Sqoop uses database table reading table row by row into HDFS. Transfer process can be customized by specifying delimiters, file formats, row ranges, columns etc. Data is imported into non-relational database Hive for query analysis. They illustrated the comparative analysis of different queries in MySQL and Hive. HiveQL requires runtime optimization in order for jobs to run efficiently and with acceptable execution times.
4 Critical Review on Sentiment Analysis Using Hadoop

Hadoop addresses all the aspects of Big Data analysis for sentiment determination. Hadoop helps collect loads of volume of data. Hadoop provides speedy data download for real-time sentiment analysis. Hadoop framework helps distribute the work among different clustering machines, thus, achieving high performance. Sentiment analysis performance is improved in Hadoop by splitting the data into modules, processing in different machines, reducing response time, and improved fault tolerance by replicating the data. Hadoop helps in collection of variety of unstructured data from multiple sources in multiple formats, across domains and efficiently processing them in multi-dimensional fashion. HiveQL can be optimized at runtime improving the execution time.

Studies show that process of sentiment analysis can be performed without compromising on accuracy and speed. It can scale to bigger data sets with better performance. Machine learning algorithms like Naïve Bayes when implemented using MapReduce gives high accuracy for large volumes of data. Machine learning algorithms provided in Mahout scales well for high-dimensional large volume and complex data and can be used in several different applications. Apache Open Source platform using Hadoop also provides reduced cost application to perform sentiment analysis thus help increasing the profit of organization.

5 Conclusion

Hadoop implemented sentiment analysis has less complex business logic implementation easily extendible and better understandability and high performance at lower cost. Table 3 shows difference between MapReduce and Hive. MapReduce or Hive UDFs help process large volume of data with accuracy and time efficiently. Machine learning algorithms implemented in Hadoop are simpler and modular with

| Table 3 | Comparitive analysis of MapReduce and hive |
|------------------|------------------------------------------|
| Sentiment classification | MapReduce | Hive |
| Hadoop component | Inner Tool |
| Built-in classifiers | N | N |
| Performance | Better Poor |
| Programming model | Java SQL |
| Data | Hierarchical/structured/unstructured Structured |
| Development time | More Less |
| Flexibility | More Less |
| Robust | More Less |
| Code branching | Easy Difficult |
| Complexity | More Less |
few lines of code. Code written in Hadoop can be easily extended. Mahout provides scalable and time efficient build-in classifier.

Hadoop supports effective sentiment analysis process for Big Data. Hadoop with it components addresses three aspects of Big Data velocity, volume, and variety. Hadoop can effectively collect data in real time from social media sites or relational database using Flume and Sqoop tool, respectively. Large data sets can be efficiently stored and retrieved from Hadoop HDFS and Hive. Hadoop supports both machine learning and lexicon-based sentiment analysis of text using MapReduce or Hive. Sentiment analysis implemented in Hadoop framework provides high accuracy with efficient processing time and lower cost.

Acknowledgements The author is grateful to (Dr.) Divakar Singh, HOD, Department of CSE, UIT, Barkatullah University, Bhopal for his valuable suggestions and comments in writing this paper. The author wish to thank Puneet Sharma, for his continuous encouragement, inspiration and guidance with his abode of experience in data mining.

References

1. The Apache Software Foundation, https://flume.apache.org
2. The Apache Software Foundation, https://sqoop.apache.org
3. Dataiku, http://www.dataiku.com/blog/2013/05/01/a-complete-guide-to-writing-hive-udf.html
4. IBM, http://www.ibm.com/developerworks/java/library/j-mahout/
5. IBM, http://www.ibm.com/developerworks/library/j-mahout-scaling/
6. Pillar, http://www.3pillarglobal.com/insights/how-to-tame-the-machine-learning-beast-with-apache-mahout
7. Rajurkar G.D., Goudar R.M.: A speedy data uploading approach for twitter trend and sentiment analysis using HADOOP. In: 2015 International Conference on Computing Communication Control and Automation, pp. 580–584. IEEE, Pune (2015)
8. Mane S.B., Sawant Y., Kazi S., Shinde V.: Real time sentiment analysis of twitter data using hadoop. In: International Journal of Computer Science and Information Technology, vol. 5(3), pp. 3098–3100, IJCST (2014)
9. Zarrad A., Aljialoud J.: The evaluation of the public opinion a case study: MERS-Cov infection virus in KSA. In: 7th International Conference on Utility and Cloud Computing, pp. 664–607. IEEE, London (2014)
10. Kim J.S., Yang M.H., Hwang Y.J., Jeon S.H., Kim K.Y., Jung I.S., Choi C.H., Cho W.S., Na J.H.: Customer preference analysis based on SNS data. In: Second International Conference on Cloud and Green Computing, pp. 609–613. IEEE, Xiangtan (2012)
11. Ramesh R., Divya G., Divya D., Kurian M.K.: Big data sentiment analysis using hadoop. In: International Journal for Innovative Research in Science & Technology, vol. 1. IJIRST (2015)
12. Hammond K., Varde A.S.: Cloud based predictive analytics. In: 13th International Conference on Data Mining Workshops, pp. 607–612. IEEE, Dallas, TX (2013)
13. Shang S., Shi M., Shan W., Hong Z.: Research on public opinion based on big data. In: 14th International Conference on Computer and Information Science, pp. 559–562. IEEE, Las Vegas, NV (2015)
14. Lui B., Blasch E., Chen Y., Shen D., Chen G.: Scalable sentiment classification for big data analysis using naive bayes classifier. In: International Conference on Big Data, pp. 99–104. IEEE, Silicon Valley, CA (2013)
15. Conejero J., Burnap P., Rana O., Morgan J.: Scaling archived social media data analysis using a hadoop cloud. In: Sixth International Conference on Cloud Computing, pp. 685–692. IEEE, Santa Clara, CA (2013)

16. Prom-on S., Ranong S.N., Jenviriyakul P., Wongkaew T., Saetiew N., Achalakul T.: DOM: a big data analytics framework for mining Thai public opinions. In: International Conference on Computer, Control, Informatics and Its Applications, pp. 1–6. IEEE, Bandung (2014)

17. Joldzic O.V., Vukovic D.R.: The impact of cluster characteristics on HiveQL query optimization. In: 21st Telecommunication Forum, pp. 837–840. IEEE, Belgrade (2013)

18. The Apache Software Foundation, https://hadoop.apache.org/