International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 06 | June-2016 www.irjet.net p-ISSN: 2395-0072

TWITTER SENTIMENT CLASSIFIER

Yellapragada Sravya[1], N.Durga Prasad[2], Yellapragada Kavya[3]

[1]U.G. Student, Dept. of C.S.E., GVP College of Engineering(Autonomous),Andhra Pradesh, India
[2]Assistant Professor, Dept. of C.S.E.,GVP College of Engineering(Autonomous),Andhra Pradesh, India
[3]U.G. Student, Dept. of C.S.E., GVP College of Engineering(Autonomous),Andhra Pradesh, India

---

Abstract - *People are always online these days buying things, reading things and watching things. They also express their opinion about the things they bought in form of reviews, comments, emails, tweets, status messages etc. All these carry information about people's opinion. Anyone who is selling a product or providing a service either online or offline need to understand what people are saying about them. This is because a brand is not what they tell the customers but it is what customers tell each other.This paper covers techniques and approaches that mines opinions and establishes its overall sentiment.*

Key Words: Sentiment Analysis, opinion mining[2], twitter, Hadoop, naive baye's, python, NLTK, regular expressions,HDFS,MapReduce.

1.INTRODUCTION:

Opinion mining also called as sentiment analysis is a field of NLP(natural language processing) that tries to extract subjective information from text. The demand for information on opinion and sentiment is always important because it decides the value of a particular company in market and helps in decision making process. The power of social media is rampant. Social media monitoring is one of the hottest topics nowadays. It is the platform to know the opinions of various people all over the world. The opinion may be on a movie, a product or a service. It is important to know whether the review is positive or negative, how people are responding to an add or an event, are people satisfied or dissatisfied with the service, how people are reacting to a candidate speech during election. All these carry information about people's opinion. Determining the polarity or the semantic orientation of document is done in sentiment analysis. Performing sentiment analysis on twitter is trickier. This is because tweets are short and contains slangs, emoticons, hashtags etc. Every second on an average 6000 tweets[1] are tweeted on twitter which corresponds to 3,50,000 tweets sent per minute, 500 million tweets per day. So we are using Hadoop framework which can process huge amounts of data fastly.

2. APPROACHES:

There are many ways of approaching this problem.As the data from twitter is Unstructured the most popular ways are Rule based approach and Machine learning based approach.

2.1 Rule based approach:

This approach look at all words in the text and classify each of them as positive or negative. If there are more positive words than negative words then classify the document as positive else negative. This would require a lexicon which is a resource in which all words have been classified as positive or negative. There are many rule based approaches suggested for sentiment analysis in which some are very complex and very good at classifying as well. Vader is one such Rule based model.

Some examples of things the rules that Vader uses are based on are:

- "I really like the new IPhone.It is really awesome" -> POSITIVE
- "I hate Grapes" -> NEGATIVE
- Use of caps,emoticons and exclamation points: "this food is amazing!!! ☺"
- Words that signal a shift in emotion like but,however etc
  Example: "I like the movie initially but after one hour it is bad"
- Adverbs that act as intensifiers like extremely,hardly,very etc.
  Example: "this restaurant is really good"

2.2 Machine Learning based approach:

This visualizes it as a classification problem and classifies a document as positive or negative.There are two methods in this approach.

- Naïve Bayes classification
- Support vector machines

Every Machine learning algorithm takes training data and test data.There are several human annotated corpora available which can be taken as training data.In order to choose the features the simplest way is look at the individual words in the document.

2.2.1 NaiveBayesclassification

If you are using Naïve Bayes thencompute posterior probabilitieslike:

$$P\left(\frac{DocisPositive}{words}\right) = \frac{P(Docispos) * P\left(\frac{W1}{Docispos}\right) * P\left(\frac{W2}{Docispos}\right) * - -}{P(w1) * P(w2) * - - -}$$

$$P\left(\frac{Docisnegative}{words}\right) = \frac{P(Docisneg) * P\left(\frac{W1}{Docisneg}\right) * P\left(\frac{W2}{Docisneg}\right) * - -}{P(w1) * P(w2) * - - -}$$

Here 'Doc' refers to 'Document', 'pos' refers to 'Positive' and 'neg' refers to 'Negative'. W1, W2,... are the words in the document. Since the denominators are same, compute the numerators and pick the class whose posterior probability is greater.

2.2.2 Support Vector Machines

Here express each document as vector and if document can be represented as [x1,x2,...xn] then words in document are [w1,w2,...wn].Each xi indicates the presence or absence of word wi.If w1 is present in the document then x1=1,else x1=0.We can also add weights to determine how positive or negative the word wi is.

2.3 HADOOP

Apache Hadoop is an open source software framework for storage and large scale processing of data-sets on clusters of commodity hardware.They provide scalability to store large volumes of data on commodity hardware.They can handle various kinds of data,so they can handle Twitter data(unstructured) efficiently.They are fault tolerant.They also have the ability to facilitate a shared environment

2.4 HDFS

The Hadoop Distributed File System, a storage system for big data. It provides two capabilities that are essential for managing big data.They are Scalability to large data sets and reliability to cope with hardware failures. HDFS achieves scalability by partitioning or splitting large files across multiple computers. This allows parallel access to very large files since the computations run in parallel on each node where the data is stored.HDFS is fault tolerant. HDFS replicates, or makes a copy of, file blocks on different nodes to prevent data loss. By default, HDFS maintains three copies of every block. HDFS is comprised of two components. NameNode, and DataNode. These operate using a master slave relationship. The NameNode is responsible for metadata and DataNodes provide block storage. The data node listens to commands from the name node for block creation, deletion, and replication.

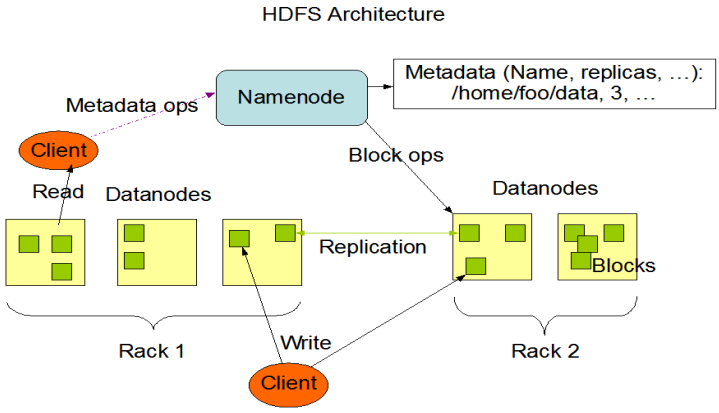

Fig-1: HDFS Architecture[5]

3. IMPLEMENTATION:

In our approach the objective is to accept a search term from the user and find the current sentiment for that term on the twitter. Here we perform sentiment analysis on big data(tweets) in the following steps and collaborate with Hadoop for storing and for performing MapReduce works. The source code can be written in Java or Python.

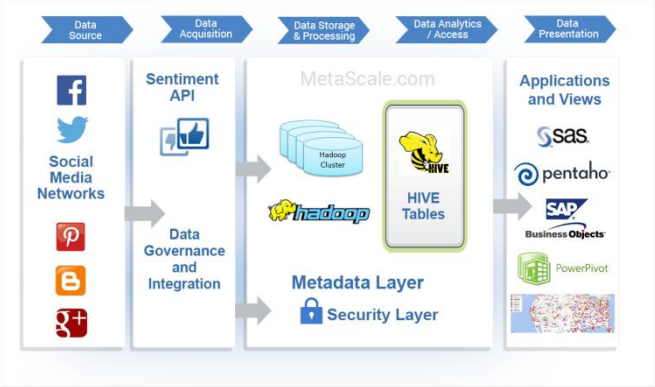

Fig-2: Sentiment Analysis Topology[4]

Step1: Accept a search term from the user and retrieve tweets for that term

Access the twitter API using python-twitter module.This can be done by registering your application on twitter and generate API key and other credentials.

Step2: Storing the retrieved data

This step is optional because Twitter has a rate limit in downloading tweets.If our rate is exceeding it,then twitter stops our access over data.So it is better to retrieve 50 or 100 tweets and process them.If we want to store tweets then data is stored in Hadoop cluster and mapping is done.

Step3:Pre Processing the data

a.Creating training data and test data sets:

Test data will be all the tweets retrieved by us and in order to form training data we should download a corpus of tweets. We will use Niek Sanders's Tweet Sentiment Corpus as CSV file who has provided 5000+ labelled tweets. While downloading use a delay to avoid being rate limited by twitter. In the corpus file each tweet is labelled as positive, negative, neutral or irrelevant. The corpus file containing tweet ID and label as twitter doesn't allow tweet text to be shared directly.

b.Pre Processing the tweets:

This can be done using regular expressions in python.The task is accomplished in a series of steps

- Convert the tweets to lower case
- Replace links with the string URL
- Replace '@' with AT_USER
- Replace '#' word with word
- For further cleanup replace repetitions of characters, for ex: change "huuuuungry" to "hungry"

Now we use NLTK(Natural Language Tool Kit) to do further Pre Processing like:

- Removing stop words(including URL and User)
- Tokenize the tweets into a list of words.

Step4: Extract the features from the training and test data:

We will do the classification in two ways. They are Naïve Bayes Classification and Support Vector Machines.

3.1 Naive Bayes Classification

Use NLTK's built in Naive Bayes classifier to train the classifier

1.Build a vocabulary (list of all the words in all the tweets in the training data)

2.Represent each tweet with the (presence/absence) of these words in the tweet

Example: {'the', 'worst', 'thing', 'in', 'the', 'world'} -> vocabulary

{'the', 'worst', 'thing'} -> tweet

(1,1,1,0,0,0) ->feature vector

3.2 Support Vector Machines

The first two steps are the same as for Naïve Bayes.To determine whether a word is positive or negative we use Lexicons.Sentiwordnet is a special lexicon that provides positive,negative and objectivity scores for every word. A Synset groups together lemma (word,meaning) pairs with the same meaning.All the elements in synset have same meaning or definition.Sentiwordnet takes the synsets and assigns them a polarity score.Every synset has three scores.A positive,negative and objectivity polarity score.These three scores add upto one.Here we will use first synset for the word.If positive score is greater than negative score then use positive score as weight, else use negative score as weight.

Example:

{'THE', 'WORST', 'THING', 'IN', 'THE', 'WORLD'} -> VOCABULARY

{'THE', 'WORST', 'THING'} -> TWEET

(0, -1, 0, 0, 0, 0) -> FEATURE VECTOR

Step5: Training classifier

Now train a classifier on training data and use the classifier to classify the problem instances.Then run the classifier on downloaded tweets, get the majority vote and print the sentiment.

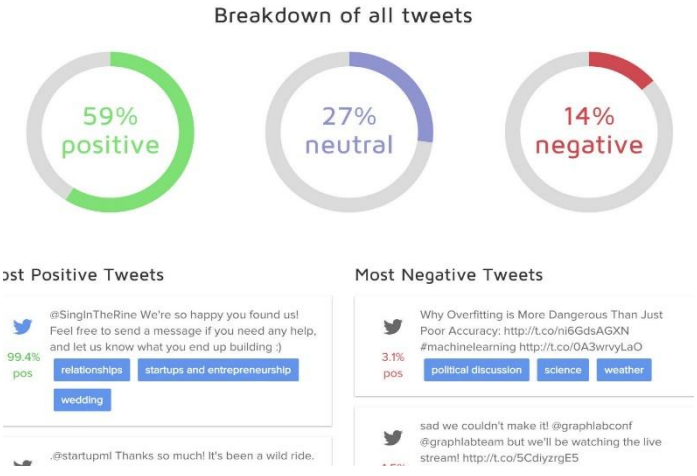

Fig-3: Result of Sentiment Analysis[3]

4. ACCURACY AND EFFICIENCY:

A lexicon based sentiment analyser produces more accurate results. The accuracy of project is determined by the time taken to produce results and also correctness in the results. As we are performing a step wise refinement the overall accuracy is purely dependent on individual

steps. Misprocessing any of the steps gives inaccurate results. we performedsentiment analysis on sanders twitter sentiment corpus which gave results like:

Table-1: Table showing accuracy in the analysis

Sentiment	Total Count	Correct--ness	%	Tolerance
Positive	815	732	89.81	-0.02
Negative	329	216	65.65	+0.05
Neutral	76	60	78.94	+-0.003

Thus the overall accuracy is 78.13.Here the time efficiency is high because we used Hadoop which uses parallel processing and distributive computing.

5. FUTURE SCOPE:

Sentiment analysis is never 100% accurate. This is because some sentences are not easy to analyse. A language is always complex to analyse. This is because of some factors like context-a word which has opposite meaning depending on context, sentiment ambiguity[6]-If a word doesn't express any kind of sentiment, sarcasm-if there is no sarcasm in sentence,comparative sentences and regional variations.Despite its drawbacks sentiment analysis is a powerful tool.

6. CONCLUSION:

Sentiment analysis is having high potential.In this project we had seen sentiment analysis as a classification problem.We had used Sander's corpus file and our accuracy is around 78%.But it is hard to classify twitter data comparing other kinds of data like reviews,mails etc.We believe that more research has to be done on this field in order to learn other reliable features and use them to improve the accuracy when running on twitter statuses.But it is the key to various predictions,analysis works and ultimately a solid insight to social performance of an enterprise.

REFERENCES:

[1]Twitter usage statistics available at http://www.internetlivestats.com/twitter-statistics/
[2]Bo Pang, Lillian Lee, "Opinion Mining and Sentiment Analysis"
[3]What's your twitter thumbprint at Indico web page https://indico.io/blog/whats-your-twitter-thumbprint/
[4]Product perception and brand sentiment analysis from metascale site http://www.metascale.com/hadoop-architecture-for-customer-perception-analysis.html
[5]HDFS Architecture guide at https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html
[6]sentiment analysis not always accurate at http://brnrd.me/sentiment-analysis-never-accurate/