International Journal of Latest Engineering Research and Applications (IJLERA) ISSN: 2455-7137

Volume – 02, Issue – 03, March – 2017, PP – 35-40

Sentimental Analysis On Twitter

Prof. Amit Narote, Sohail Shaikh, Saville Pereira, Nitin Jadhav, Platini Rodrigues

XAVIER INSTITUTE OF ENGINEERING

Mahim causeway, Opp.Raheja Hospital, Mumbai-400016.

Abstract: In today's world with the rise of social networking approach, there has been a surge of user generated content. There are a large number of social media websites that enable users to contribute, modify and grade the content. Users have an opportunity to express their personal opinions about specific topics. The example of such websites include blogs, forums, product reviews sites, and social networks. In this case, social media data is used. Sites like twitter contain prevalently short comments, like status messages called as "tweets" on social networks like twitter. Additionally many web sites allow rating the popularity of the messages which can be related to the opinion expressed by the author. The focus of project is to analyze the polarity to tweet's or post i.e. what is trends going on now days. Micro blogging sites have millions of people sharing their thoughts daily because of its characteristic short and simple manner of expression. We propose and investigate a paradigm to mine the sentiment from a popular real-time micro blogging service, Twitter, where users post real time reactions to and opinions about "everything". In this project, we expound a hybrid approach using both corpus based and dictionary based methods to determine the semantic orientation of the opinion words in tweets. A case study is presented to illustrate the use and effectiveness of the proposed system.

Index Terms: Big Data; data mining; public opinion; Hadoop; Mahout

1. Introduction

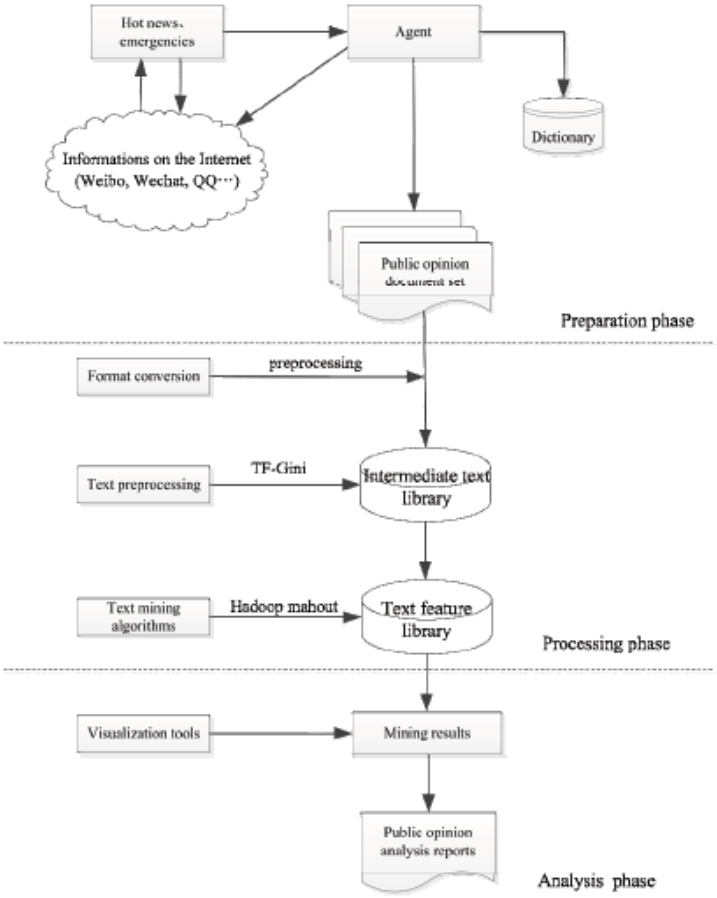
On-going increase in wide-area network connectivity promise vastly augmented opportunities for collaboration and resource sharing. Now-a-days, various social networking sites like Twitter, Facebook, MySpace, and YouTube have gained so much popularity and we cannot ignore them. They have become one of the most important applications of Web 2.0 [1]. They allow people to build connection networks with other people in an easy and timely way and allow them to share various kinds of information and to use a set of services like picture sharing, blogs, wikis etc.

It is evident that the advent of these real-time information networking sites like Twitter have spawned the creation of an unequalled public collection of opinions about every global entity that is of interest. Although Twitter may provision for an excellent channel for opinion creation and presentation, it poses newer and different challenges and the process is incomplete without adept tools for analysing those opinions to expedite their consumption.

More recently, there have been several research projects that apply sentiment analysis to twitter corpora in order to extract general public opinion regarding political issues [2]. Due to the increase of hostile and negative communication over social networking sites like Facebook and Twitter, recently the Government of India tried to allay concerns over censorship of these sites where Web users continued to speak out against any proposed restriction on posting of content. As reported in one of the Indian national newspaper [3] "Union Minister for Communications and Information Minister, Kapil Sibal, pro-posed content screening & censorship of social networks like Twitter and Facebook". Instigated by this the research carried out by us was to use sentiment analysis to gauge the public mood and detect any rising antagonistic or negative feeling on social medias. Although, we firmly believe that censorship is not right path to follow, this recent trend for research for sentiment mining in twitter can be utilized and extended for a gamut of practical applications that range from applications in business (marketing intelligence; product and service bench marking and improvement), applications as sub-component technology (recommender systems; summarization; question answering) to applications in politics. This motivated us to propose a model which retrieves tweets on a certain topic through the Twitter API and calculates the sentiment orientation/score of each tweet.

The area of Sentiment Analysis intends to comprehend these opi-nions and distribute them into the categories like positive, negative, neutral. Till now most sentiment analysis work has been done on review sites [4]. Review sites provide with the sentiments of products or movies, thus, restricting the domain of application to solely business. Sentiment analysis on Twitter posts is the next step in the field of sentiment analysis, as tweets give us a richer and more varied resource of opinions and sentiments that can be about anything from the

latest phone they bought, movie they watched, political issues, religious views or the individuals state of mind. Thus, the foray into Twitter as the corpus allows us to move into different dimensions and diverse applications.



2. System Architecture

2.1 Preparation Phase

1. The agent read keywords from dictionary which contains the public opinion hot topics key words.
2. According the key words, the agent search on the Weibo, Wechat, and QQ group to download the hot pot messages.
3. All the messages will be grouped into document sets.

2.2 Processing phase

1. Format conversion: The initial message most probable html or other format. The format conversion's function is to convert the initial messages into text format.
2. After the format conversion, we can use TF-Gini algorithm To select feature words which will be stored into the intermidiate text library.
3. To use the Hadoop Mahout Text mining algorithms process the public opinion messages.This is the most important step in the system. The mining results will be stored into the text feature library.

2.3 Analysis Stage

1. Visualization tools read mining results from the text feature library.
2. According to the results, visualization tools generate various forms of reports, such as histograms, pie charts, graphs and so on.
3. According to the reports, the decier's decide what or how to do next.

In this paper, we mainly consider the TF-Gini and Hadoop Mahout Text mining algorithms. Other processing steps using the third party open source processing framework.

3. The Core Algorithms

A. TF-Ginialgorithm

The original Gini algorithm formula is:

$$Gini(Q) = 1 - \sum_{i=1}^{|C|} P_i^2 \quad (1)$$

Where, *Q* is a set of samples, *Pi* is the probability for samples set *Qi* to belong to *Ci*, |*C*| is the total amount of classes. The original Gini Index is used to measure the classification attributes' "complexity". Note that, a smaller "complexity" means a better attribute. The formula can be described as follows:

$$Gini(Q) = \sum_{i=1}^{|C|} P_i^2 \quad (2)$$

A bigger "purity" means that the attributes have more abundant information. "Purity" reflects the influence of feature selection in classification. This paper uses "purity" to measure the Gini algorithm in feature selection.

The TF-IDF algorithm is a classical algorithm to calculate the features' weights. For a word *w* and a text *d*, the weight of *w* in *d* is as follows:

$$TFIDF(d, w) = TF(d, w) \cdot IDF(d, w) = TF(d, w) \cdot \log\left(\frac{|D|}{DF(w)}\right) \quad (3)$$

Where, *TF(d,w)* is the frequency that *w* appears in text *d*. |*D*| is the number of classes. *DF(w)* is the number that w appears in the training set.

However, the TF-IDF algorithm is not suitable for text classification, mainly because of the shortcoming of TF-IDF in the section of IDF. The TF-IDF considers that a word with a higher frequency in different texts in the training set is not important. This consideration is not appropriate for text classification. We use purity Gini to replace the IDF part in the TF-IDF formula. It can be described as follows：

$$GiniTxt(w) = \sum_{i=1}^{|C|} \frac{P(w|c_i)}{P(w|c_i)} \quad (4)$$

B. The Mahout clustering algorithms

1. Canopy algorithm

The main idea of the Canopy algorithm is divided clustering algorithm into two stages. 1) By using a simple
distance computing method, to divide the data sets into overlapping subsets – canopy.2) By using a precise and rigorous computing method to calculate the data distance vectors in the same Canopy.

The difference between Canopy algorithm and traditional clustering algorithm is the Canopy using two computing distance methods and only computing the overlapping data vectors. The step of creating canopy is as follows:

1). Suppose *List* is the initial data collection, and the initial distance threshold is T_1, T_2 ($T_1 > T_2$).

2). Picking a data vector *A* randomly from *List*, using a simple computing method to calculate the distance *d* between vector *A* and the other samples in the *List*.

3). If $d < T_1$, put the samples into the same canopy. Meanwhile, if $d < T_2$, remove the samples from *List*.

4). Repeat step2 and step3, until the *List* is empty.

2. K-Means algorithm

K-Means algorithm is a widely used clustering algorithm. Randomly select *k* objects, each object represents a cluster center. For the remaining objects, according to their distance to the center of each cluster, divide them

into the smallest distance cluster center, and then re-calculate each cluster center. Repeat this process until the clustering criterion function converges. Criterion function has two forms:

1). Global tolerance function

$$E = \sum_{i=1}^{k} \sum_{x_j \in S_i} (x_j - \mu_i)^2 \qquad (5)$$

Where, E is the convergence criteria. If E is less than the threshold, the iterative process will be end. Otherwise, continue to the next iteration process. k is cluster center, S_i is one of the k group, μ_i is S_i's center, x_j is the sample in S_i.

2). tolerance changes

$$E = \sum_{i=1}^{k} (\mu_{ib} - \mu_{ia})^2 \qquad (6)$$

Where, E has the same meaning with formula *(5)*. k is cluster center, μ_{ib} is the previous center, , μ_{ia} is the last center.

The step of K-Means algorithm is as follows:
1). randomly select k objects from data set D as initial cluster centers.

2). According to the distance, divide the remaining objects into the smallest distance between the sample and cluster center.

3). Update the cluster centers, recalculate the cluster centers.

4). Computing the criterion function.

5). If meet the criteria function, exit. Otherwise go to step 2.

C. The Mahout Classification algorithm

Naïve Bayes algorithm is a popular algorithm in text classification field. Naïve Bayes algorithm assumes that the value of each feature is independent and this assumption called condition independence that used to simplify the computation, and in this case, we call it "Naïve". Naïve Bayes is based on Bayesian theorem.

It assumes that d is a data sample with unknown class label and H' is an assumption. If data sample d belongs to a particular class c, for the problem of categorization, we hope to get $P(H' \mid d)$. Namely, we hope to know the probability of H' when data sample d is given. $P(H' \mid d)$ is a posteriori probability or a posteriori probability under the condition of d.

AS we know that, $P(d)$, $P(H')$ and $P(H' \mid d)$ an be calculated from the given data. The Bayesian theorem provides a method for calculated from the given data. So, Bayesian theorem can be described as follows:

$$P(H' \mid d) = \frac{P(d \mid H')}{P(d)} \qquad (7)$$

Each data sample is represented as an n-dimensional feature vector that describes n measures of n samples. Assumed m class of $c_1, c_2, \ldots, c_m$ and given an unknown data sample d (no class label), they will be sorted into the class which has the highest posteriori probability based on categorization. In other words, a native Bayesian classifier will assign unknown sample to the class c_i, , iff : $P(c_i \mid d) > P(c_j \mid d), 1 \le i, j \le m, j \ne i$.
Thus, we can maximize the $P(c_i \mid d)$, where class c_i has the largest $P(c_i \mid d)$ and is called the maximum posteriori assumption. According to Bayesian theorem:

$$P(c_i \mid d) = \frac{P(d \mid c_i)}{P(d)} \qquad (8)$$

Since $P(d)$ is a constant for all classes, we only need to maximize $P(d|c_i)\ P(c_i)$. If the prior probability of the class is unknown, it is usually assumed that the probability of these class is equivalent, that is $P(c_1) = P(c_2) = ... = P(c_m)$. So we only need to maximize $P(d|c_i)$. Otherwise, we should maximize The $P(d|c_i)\ P(c_i)$ .We also know that the prior probability of a class can be calculated by $P(c_i) = s/s^i$ where s_i is the number of training samples of the class and s is the total number of training samples. It may cost much time to calculate $P(d|c_i)$ when the given data sets with many attributes. To reduce the computational cost of $P(d|c_i)$, we can simply assume that the class is conditional independent. If we know the class label of a sample, and assume that the value of each property is conditional independent, namely, there is no dependent relationship between every pair of properties. Therefore $P(d|c_i)$ can be calculated as follows:

$$P(d|c_i) = \prod_{j=1}^{n} P(x_j|c_i) \qquad (9)$$

C. Pattern mining algorithm

Apriori is an algorithm for frequent item set mining and association rule learning over transactional databases. It proceeds by identifying the frequent individual items in the database and extending them to larger and larger item sets as long as those items often appear sufficiently in the database. Apriori uses breadth-first search and a Hash tree structure to count candidate item sets efficiently. It generates candidate sets of length *k* from item sets of length *k-1*. Then it prunes the candidates which have an infrequent sub pattern. The pseudo code for the algorithm is given below for a transaction database *T*, and a support threshold of ∈.

Apriori $(T, \in)$
  $L_1 \leftarrow \{large1\text{-}itemsets\}$ $k \leftarrow 2$
  while $L_{k-1} \neq \emptyset$
    $C_k \leftarrow \{a \cup \{b\} \mid a \in L_{k-1} \wedge b \in \cup L_{k-1} \wedge b \notin a\}$ for *transactions* $t \in T$
      $C_t \leftarrow \{c \mid c \in C_k \wedge c \subseteq\}$ for *candidates* $c \in C_t$
        $count[c] \leftarrow count[c] + 1$
    $L_k \leftarrow \{c \mid c \in C_k \wedge count[c] \geq \in\}$ $k \leftarrow k + 1$
return $\cup_k L_k$

FP-tree algorithm is an alternative way to find frequent item sets without using candidate generations. The core of this method is the usage of a special data structure named frequentpattern tree (FP-tree), which retains the item set association information.

The FP-tree algorithm works as follows:

First it compresses the input database creating an FP-tree instance to represent frequent items. Then it divides the compressed database into a set of conditional databases, each one is associated with one frequent pattern. Finally, each database is mined separately.

In Mahout, we use Parallel Frequent Pattern Mining (PFPM). This algorithm contains 5 steps: 1) Finding the onedimensional frequent items from the original data. 2) According to the one-dimensional frequent items, divide the original data set into different groups. 3) Creating FP-tree on each groups. 4) Mining the frequent items on each FP-tree. 5) Merging each FP-tree frequent items results.

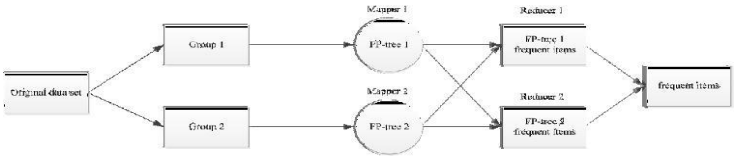
4. Conclusion

The proliferation of micro blogging sites like Twitter offers an unprecedented opportunity to create and employ theories & technologies that search and mine for sentiments. The work presented in this paper specifies a novel approach for sentiment analysis on Twitter data. To uncover the sentiment, we extracted the opinion words (a combination of the adjectives along with the verbs and adverbs) in the tweets. The corpus-based method was used to find the semantic orientation of adjectives and the dictionary-based method to find the semantic orientation of verbs and adverbs. The overall tweet sentiment was then calculated using a linear equation which incorporated emotion intensifiers too. This work is exploratory in nature and the prototype evaluated is a preliminary prototype. The initial results show that it is a motivating technique.

5. Acknowledgment

We would like to thank Fr. Francis de Melo (Director of XIE) for providing us with such an environment so as to achieve goals of our project and supporting us constantly.We express our sincere gratigratitude to our Honourable Principal Dr.Y.D.Venkatesh for encouragement and facilities provided to us.We would like to place on record our deep sense of gratitude to Head of Dept. Of Information Technology, Xavier Institute of Engineering,

Mahim, Mumbai, for her generous guidance help and useful suggestions.With deep sense of gratitude we acknowledge the guidance of our project guide Prof. Amit Narote. The time-to-time assistance and encouragement by her has played an important role in the development of our project.We would also like to thank our entire Information Technology staff who have willingly co-operated with us in resolving our queries and providing us all the required facilities on time



6. REFERENCES

[1] L. Colazzo, A. Molinari and N. Villa. "Collaboration vs. Participation: the Role of Virtual Communities in a Web 2.0 world", International Conference on Education Technology and Computer, 2009, pp.321-325.

[2] nlp.stanford.edu/courses/cs224n/2011/reports/patlai.pdf

[3] National Daily, Economic Times: articles.economictimes.indiatimes.com › Collections › Facebook

[4] K. Dave, S. Lawrence and D.M. Pennock. "Mining the peanut gallery: Opinion extraction and semantic classification of product reviews". In Proceedings of the 12th International Conference on World Wide Web (WWW), 2003, pp. 519–528.

[5] A. Pak and P. Paroubek. "Twitter as a Corpus for Sentiment Analysis and Opinion Mining".In Proceedings of the Seventh Conference on International Language Resources and Evaluation, 2010, pp.1320–1326.

[6] R. Parikh and M. Movassate, "Sentiment Analysis of User- Generated Twitter Updates using Various Classification Techniques", CS224N Final Report, 2009

[7] A. Go, R. Bhayani, L.Huang. "Twitter Sentiment Classification Using Distant Supervision". Stanford University, Technical Paper ,2009

[8] L. Barbosa, J. Feng. "Robust Sentiment Detection on Twitter from Biased and Noisy Data". COLING 2010: Poster Volume, pp. 36-44.

[9] S. Batra and D. Rao, "Entity Based Sentiment Analysis on Twitter", Stanford University,2010

[11] A. Bifet and E. Frank, "Sentiment Knowledge Discovery in Twitter Streaming Data", In Proceedings of the 13th International Conference on Discovery Science, Berlin, Germany: Springer,2010, pp. 1–15.

[12] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, R. Passonneau, "Sentiment Analysis of Twitter Data", In Proceedings of the ACL 2011 Workshop on Languages in Social Media,2011 , pp. 30–38

[13] A. Kumar. and T. M. Sebastian, "Sentiment Analysis: A Perspective on its Past, Present and Future", International Journal of Intelligent Systems and Applications (IJISA), MECS Publisher, 2012 (Accepted to be published)

[14] A. Kumar and T. M. Sebastian, "Machine learning assisted Sentiment Analysis". Proceedings of International Conference on Computer Science & Engineering (ICCSE'2012), 2012, pp. 123-130.

[15] POS Tagger: http://www.infogistics.com/textanalysis.html

[16] V. Hatzivassiloglou and K. McKeown, "Predicting the semantic orientation of adjectives". In Proceedings of the Joint ACL/EACL Conference,2004, pp. 174–181

[17] WordNet: http://wordnet.princeton.edu/

[18] Songtao Shang; Minyong Shi; Wenqian Shang; Zhiguo Hong, "Research on public opinion based on Big Data," in Computer and Information Science (ICIS), 2015 IEEE/ACIS 14th International Conference on , vol., no., pp.559-562, June 28 -July 1 2015

[19] Yaxiong Zhao; Jie Wu; Cong Liu, "Dache: A data aware caching for big-data applications using the MapReduce framework," in Tsinghua Science and Technology , vol.19, no.1, pp.39-50, Feb. 2014

[20] Segev, A.; Chihoon Jung; Sukhwan Jung, "Analysis of Technology Trends Based on Big Data," in Big Data (BigData Congress), 2013 IEEE International Congress on , vol., no., pp.419-420, June 27 2013-July 2 2013

[21] Kaisler, S.; Armour, F.; Espinosa, J.A.; Money, W., "Big Data: Issues and Challenges Moving Forward," in System Sciences (HICSS), 2013 46th Hawaii International Conference on , vol., no., pp.995-1004, 7-10 Jan. 2013

[22] Wenbo Wang; Lu Chen; Thirunarayan, K.; Sheth, A.P., "Harnessing Twitter "Big Data" for Automatic Emotion Identification," in Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom) , vol., no., pp.587-592, 3-5 Sept. 2012.

