International Journal of Innovative Technologies

WWW.IJITECH.ORG

ISSN 2321-8665
Volume.06, Issue.01,
January-June, 2018,
Pages:0114-0116

Twitter Sentiment Analysis on Twitter Data using R

K. SAI DILIP REDDY[1], SHAIK REHMATHUNNISA NAGA[2]

[1]PG Scholar, Dept of CSE, DJR College of Engineering & Technology, AP, India, E-mail: dilip14242@gmail.com.

[2]Associate Professor, Dept of CSE, DJR College of Engineering & Technology, AP, India, E-mail: shaikrehmathunnisa@gmail.com.

Abstract: Twitter a free micro blogging tool allow people to express their point of view on a wide range of topics ranging from marketing to customer service which can be used for sentiment analysis in order to identify customer likes, opinions, dislikes, feedback, etc. The use of natural language processing, text analysis and computational linguistics which involves identifying people's opinion in a given data is known as sentiment analysis also known as opinion mining. Social media plays a significant role in sentiment analysis which can be used for better decision-making approach. This paper introduces a lexicon-based approach to classify the tweets on #keywords in terms of positive, negative and neutral polarity. A shiny dashboard web application is been created to visualize the data. The aim of the paper is to discover what the people's sentiment on a keyword e.g.: demonetization and thus to present the results of analysis in the shiny dashboard.

Keywords: Sentiment Analysis, Opinion Mining, Twitter, R Programming, Shiny Dashboard, Shiny.

I. INTRODUCTION

Social networks have revolutionised the way in which people communicate. Information available from social networks is beneficial for analysis of user opinion, for example measuring the feedback on a recently released product, looking at the response to policy change or the enjoyment of an ongoing event. Manually sifting through this data is tedious and potentially expensive. Sentiment analysis is a relatively new area, which deals with extracting user opinion automatically. An example of a positive sentiment is, “natural language processing is fun” alternatively, a negative sentiment is “it’s a horrible day, I am not going outside”. Objective texts are deemed not to be expressing any sentiment, such as news headlines, for example “company shelves wind sector plans”. There are many ways in which social network data can be leveraged to give a better understanding of user opinion such problems are at the heart of natural language processing (NLP) and data mining research. In this paper we present a twitter data and sentiment analysis which is able to analyse Twitter keyword. We show how to automatically collect a real-time data from twitter for sentiment analysis and opinion mining purposes. On those tweets we apply lexicon-based approach, that is able to determine positive, negative and objective sentiments for a document.

II. APPROACH

Tweets related of #keywords are been extracted from the twitter using twitterR package. A sequence of strings been broken down into pieces like keywords, words, symbols, phrases and other elements called tokens is the job of tokenization. Often punctuations are been removed in this process. A polarity score is then been assigned to each of the element. In order to determine the sentiment behind the text the aggregated sum of the score is been calculated. Depending on the calculated score the text is been classified as positive, negative and neutral.

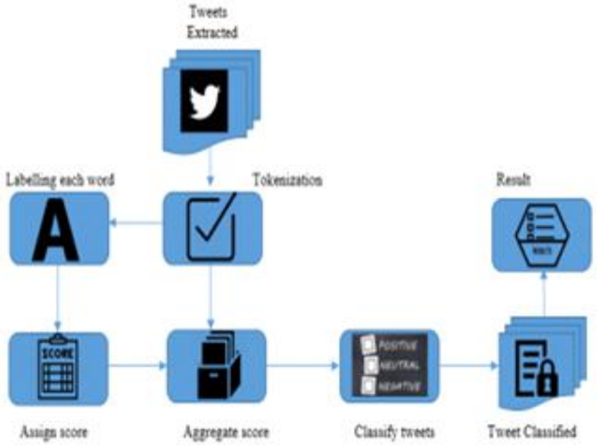

Fig.1.

III. METHODOLOGY

A. Planning and Data Collection

An easy way to extract tweets containing a hashtag say #keyword from a user account or public tweets, the twitteR package is been introduced. The search Twitter function can be used to retrieve tweets been tweeted for #keyword. Hashtags are basically used to categorize your tweets making it easy to search. An app needs to be created on dev.twitter.com, before loading twitteR library and using its functions. The Search API could produce only 1500 tweets at a time which is one of the constraints imposed by Twitter. Max of 1500 tweets related to #keyword were gathered, to carry out sentiment analysis using R programming language. The total number of tweets extracted from twitter were merged and stored in a csv file named keyword.csv.

B. Data Pre-Processing

First step is to convert all the tweets been extracted into lower case using to lower function. Lower-case is suitable if the hashtag is of one word or a shorter version. Twitter however doesn't distinguish between cases while a search is been carried. For example: Search for #happy and #happy while return the same result. Once the tweets are been converted into lowercase, the next step is to remove punctuations using remove Punctuation function in the tm package. Using multiple punctuations and spaces will result in your hashtag making no sense. Stop words are a set of commonly used words not only in English but in any other language. In order to concentrate on important words instead of very commonly used in a given language, stop words are been removed. For tweets, terms like "#", "RT", "@username" can be likely regarded as stop words. For example: If we search for "how to develop a web application" the search engine will show up with multiple pages containing the terms "how", "to", "a" instead of main keywords "develop","web", "applications". Just for simplicity, tweets consisting of numbers are been removed. URL's are not an essential element to be considered in the case of sentimentanalysis, hence removed. Also, a long URL consumes a huge portion of allocated 140 characters.

C. Bags of Words

The baseline approach also known as "Bag of Words Approach" is the simplest and most widely used lexicon based approach. This method consists of two dictionaries – one of the positively tagged words and other of negatively tagged words. After tokenization, search is carried for each individual word of the tweet within those dictionaries, and a polarity score is been assigned depending upon the location of the word. Consider a tweet: "Demonetization is good in the long run and will bring such a positive change for Indian economy". At the end of data pre-processing, the resultant text for analysis is – "demonetization is good in the long run and will bring such a positive change for Indian economy". According to the technique explained above, the words - "good"," positive" are allocated a sentiment score of +1 since they are present in the dictionary of positive word. The total polarity score of +2 is obtained on aggregation, indicating the sentiment behind the tweet as positive.

Scoring: If an individual token is been found in the dictionary of positive words, it is assigned a polarity score of +1 and if present in the dictionary of negative words, a score of -1 is been assigned, else a score of 0 is assigned.

Aggregation: The total sum of the scores allocated to an individual word in the text is calculated and based on the final polarity value, tweets can be categorized as positive, neutral or negative.

D. Shiny Dashboard

Application Shiny/Shiny dashboard is a package that enables you to easily create flexible, attractive, interactive dashboards with R. Shiny dashboard package can be easily installed from CRAN (Comprehensive R Archival Network). Shiny dashboard is a standard HTML document, hence can be deployed on any web server. For additional interactivity, Shiny components can be added and then deployed on your own Shiny Server or shinyapps.io. The basic structure of Shiny dashboard application is separated in a user-interface script and server script file. The layout and appearance of the application is been controlled by user-interface (ui), defined in a source script named ui.R. Instructions that the computer requires building an application is contained in the server. R script. A dashboard build with shiny dashboard consists of three main elements

- The dashboard header
- The dashboard sidebar
- The dashboard body

The Shiny and Shiny dashboard package are designed primarily to run applications locally and are free and open source. Once all the unwanted features are been removed during data pre-processing, a time series plot can be prepared out of the final pre-processed data. Time series plot helps to make the comparative study of the sentiments on day to day basis. consists of the classified tweets along with the sentiment score as per day. The visual representation is created using time series plot. It consists of x and y axis with various date and time. The graphical representation of the distribution of tweets is shown on the graph. Pie-Chart provides us the exact percentage of people sources from where tweets are mostly collected. From the analysis carried out it is clearly stated that people do not tend to be much in favour of mobile sources.

IV. CONCLUSION

In this paper we have presented a way how a machine learning techniques can be applied to twitter data to establish membership, in this case positivity, negativity and objectivity. We have looked at common process in NLP that can help us derive the meaning or context of a given phrase. We have demonstrated how to collect an original corpus for sentiment classification and the refinement that is needed with such data. We have applied a lexical technique to this set conduct sentiment analysis and have found this process to be successful. On analysis of our results we have confirmed that sentiment analysis enables one to understand public sentiments with respect to specific products/services by their Comments and feedback and hence can be used for better decision-making approach.

V. REFERENCES

[1] D. O. Computer, C. wei Hsu, C. chung Chang, and C. jen Lin. A practical guide to support vector classification chih-weihsu, chih-chungchang, and chih-jen lin. Technical report, 2003.

[2] N. Cristianini and J. Shawe-Taylor.An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge University Press, March 2000.

[3] B. J. Jansen, M. Zhang, K. Sobel, and A. Chowdury. Micro-blogging as online word of mouth branding. In CHI EA '09: Proceedings of the 27th international conference extended abstracts on Human factors in computing systems, pages 3859–3864, New York, NY, USA, 2009. ACM.

[4] T. Joachims. Making large-scale support vector machine learning practical. In B. Sch¨olkopf, C. J. C. Burges, and A. J. Smola, editors, Advances in kernel methods: support vector learning, pages 169–184. MIT Press, Cambridge, MA, USA, 1999.

[5] C. D. Manning and H. Schutze. Foundations of statistical natural language processing.MIT Press, 1999.

[6] G. Mishne. Experiments with mood classification in blog posts. In 1st Workshop on Stylistic Analysis Of Text For Information Access, 2005.

[7] K. Nigam, J. Lafferty, and A. Mccallum.Using maximum entropy for text classification. In IJCAI-99 Workshop on Machine Learning for Information Filtering, pages 61–67, 1999.

[8] B. Pang and L. Lee.Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval, 2(1-2):1–135, 2008.

[9] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? Sentiment classification using machine learning techniques. In Proceedings of the Conference on
Empirical Methods in Natural Language Processing (EMNLP), pages 79–86, 2002.

[10] J. Read. Using emoticons to reduce dependency in machine learning techniques for sentiment classification.In Proceedings of ACL-05, 43nd Meeting of the Association for Computational Linguistics.Association for Computational Linguistics, 2005.

[11] T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis.In Proceedings of Human Language Technologies Conference/ Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP 2005), Vancouver, CA, 2005.

Author's Profile:

K. Sai Dilip Reddy, completed his B.Tech in Computer Science And Engineering and pursuing M.Tech in Computer Science And Engineering in DJR College of Engineering and Technology.

ShaikRehmathunnisa Naga,M.Tech received her M.Tech degree and B.Tech degree in Computer Science And Engineering. She is currently working as an Assoc Professor in , DJR College of Engineering & Technology.