



Sentimental Analysis of Twitter Data using Python Script

Shallu Bashambu¹, Varunika Gupta², Neil Takru³

¹Assistant Professor, Maharaja Agrasen Institute of Technology, Delhi

^{2,3}Student, Maharaja Agrasen Institute of Technology, Delhi

ABSTRACT

Sentimental Analysis is a process of collecting and analyzing data based upon the person's feelings, reviews, and thoughts. Sentimental analysis often called opinion mining as it mines the important feature of people opinions. It helps analyze data into positive, negative or neutral sentiment. Using Twitter API and python packages, sentiments are computed and the result is depicted graphically.

Keywords: Sentimental Analysis, Opinion Mining.

1. INTRODUCTION

A basic task in sentiment analysis is classifying the polarity of a given text at the document, sentence, or feature/aspect level—whether the expressed opinion in a document, a sentence or an entity feature/aspect is positive, negative, or neutral. Advanced, "beyond polarity" sentiment classification looks, for instance, at emotional states such as happy, sad etc.

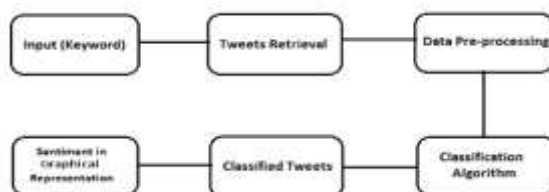


Figure 1 Proposed System

Using Twitter API and tweepy, twitter data is extracted into a relational table created using panda or numpy. Now, this data is having a lot of garbage data which needs to be removed. After the cleaned data, this data is tokenized to compute polarity of each word which is then used to compute the final result.

This project can be used by media houses to show the world about the review i.e. how people took any given situation in a positive, negative or neutral way.

2. TWITTER EXTRACTION

Using OAuth tool and tweepy, connection with twitter is made by using consumer key, consumer secret key, access token and access token secret. This extracts twitter data in JSON format which is then converted into human-readable form. All this data is handled using panda which provides an easy way to do data analysis. It provides a data structure which gives a relation view to it.

3. CLEAN-UP OF TWEETS

NLTK is used to remove stop words like a, an, the etc. which has no meaning while computing sentiments. This is done by tokenizing each tweet and then classifying them into stop words and nonstop words which are done by comparing each token using a database already present in NLTK.

4. SENTIMENTAL ANALYSIS

After computing these meaningful tokens, textblob is used to access common text processing operations which help in computing sentiments. It consists of a database where each token can be found having its value 1, 0 or -1 for positive, negative or neutral sentiment respectively. A sum of these entire token is computed related to a particular tweet to compute the final result.

5. GRAPHICAL REPRESENTATION

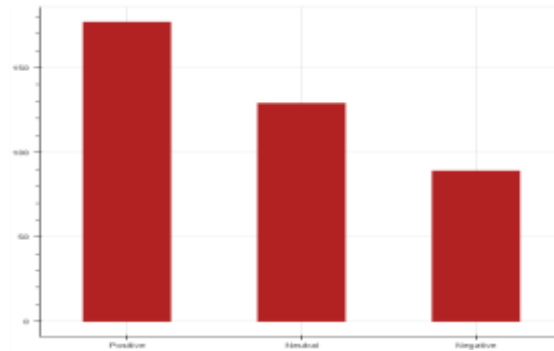


Figure 2 Bar graph

Above graph shows the comparison of a number of positive, neutral and negative tweets which is plotted using bokeh which is a python visualization library that targets modern browsers for web browsers.

6. GRAPHICAL USER INTERFACE

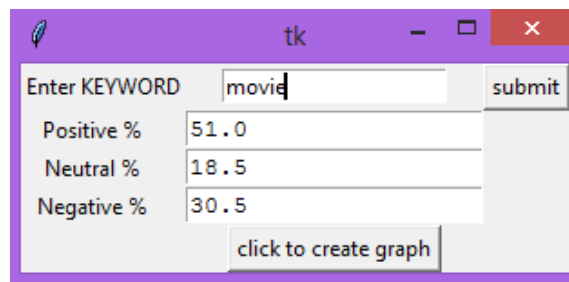


Figure 3 GUI

A dialogue box is given to the user where the keyword can be mentioned, on which sentimental analysis is to be done. Percentage of all three sentiments is computed and a button is given to produce the graphical representation.

7. REFERENCES

- [1] Briand, L. C., Daly, J., and Wüst, J., "A unified framework for coupling measurement in object-oriented systems", IEEE Transactions on Software Engineering, 25, 1, January 1999, pp. 91-121.
- [2] Maletic, J. I., Collard, M. L., and Marcus, A., "Source Code Files as Structured Documents", in Proceedings 10th IEEE International Workshop on Program Comprehension (IWPC'02), Paris, France, June 27-29 2002, pp. 289-292.
- [3] Marcus, A., Semantic Driven Program Analysis, Kent State University, Kent, OH, USA, Doctoral Thesis, 2003.
- [4] Marcus, A. and Maletic, J. I., "Recovering Documentation-to-Source-Code Traceability Links using Latent Semantic Indexing", in Proceedings 25th IEEE/ACM International Conference on Software Engineering (ICSE'03), Portland, OR, May 3-10 2003, pp. 125-137.
- [5] Salton, G., and Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer, Addison-Wesley, 1989.