School of Information Sciences

(A Constituent Institute of Manipal University)

Sentimentalytics

Twitter: Social Media and Text Analysis

Surya S	**171046022**
Eldridge Gomes	**171046003**

Course: Master of Engineering - Big Data and Data Analytics

Guide : Deepak Rao

Index

- Abstract
- Introduction
- Specification
- Design
  - Flowchart
  - Architecture
  - Module Description
- Results
- Scope for further work
- Individual Contribution

Abstract

Sentiment analysis and opinion mining is the field of study that analyses people's opinions, sentiments, evaluations, attitudes, and emotions from written language. It is one of the most active research areas in natural language processing and is also widely studied in Data Mining, Web mining, and Text Mining. In fact, this research has spread outside of computer science to the management sciences and social sciences due to its importance to business and society as a whole. The growing importance of sentiment analysis coincides with the growth of social media such as reviews, forum discussions, blogs, micro-blogs, Twitter, and social networks. For the first time in human history, we now have a huge volume of opinionated data recorded in digital form for analysis. Sentiment analysis systems are being applied in almost every business and social domain because opinions are central to almost all human activities and are key influencers of our behaviours. Our beliefs and perceptions of reality, and the choices we make, are largely conditioned on how others see and evaluate the world. For this reason, when we need to make a decision we often seek out the opinions of others. This is true not only for individuals but also for organizations.

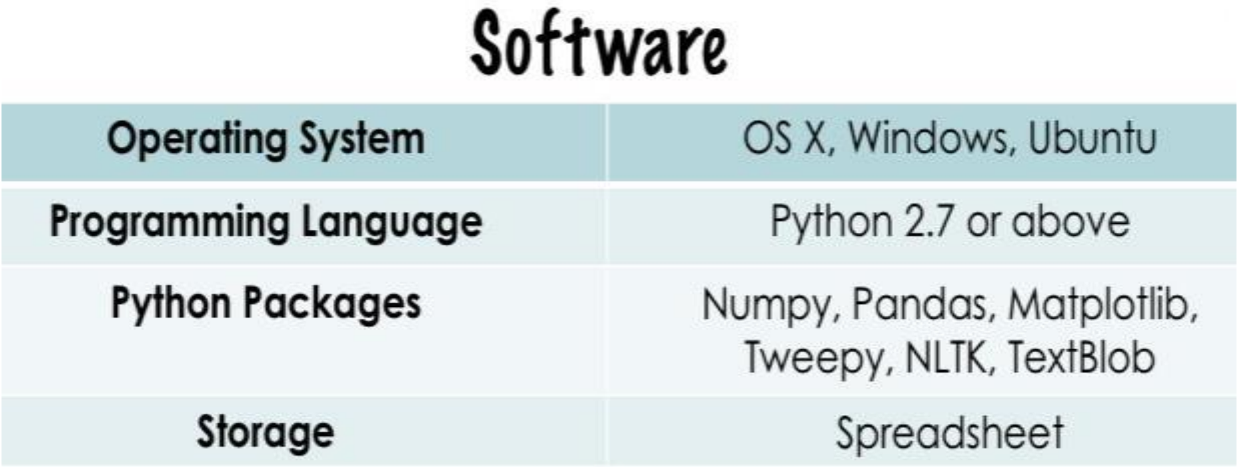
Introduction

Social media have received more attention nowadays. Public and private opinion about a wide variety of subjects are expressed and spread continually via numerous social media. Twitter is one of the social media that is gaining popularity. Twitter offers organizations a fast and effective way to analyse customers perspectives toward the critical to success in the market place. Developing a program for sentiment analysis is an approach to be used to computationally measure customers perceptions.

The objective of this project is to analyse the sentiment of twitter users through their tweets to understand whether the reaction of the people is positive, negative or neutral towards a particular event.

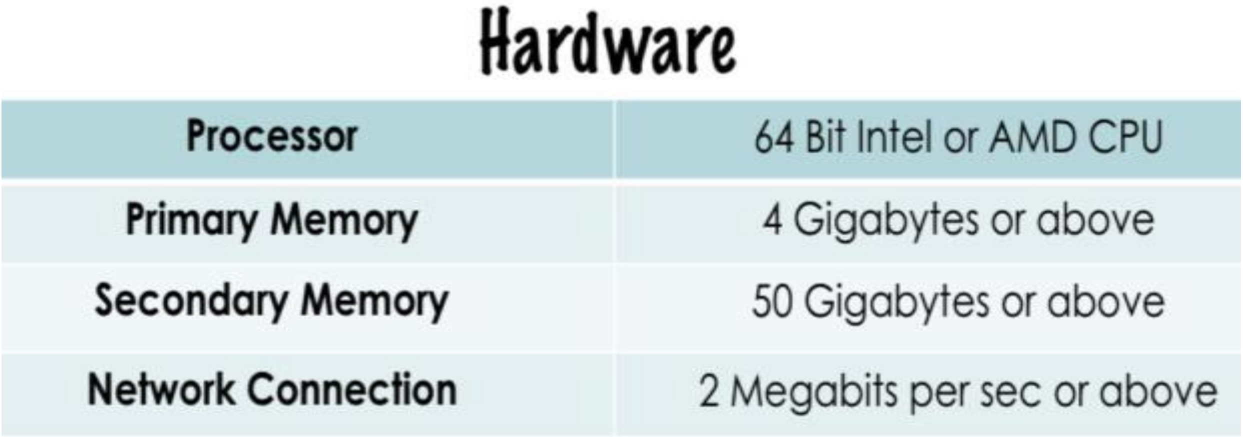
Tweets are retrieved using twitter’s Streaming APIs which is a push of data as tweets happen in near real-time. Users register a set of criteria (keywords, usernames, locations) and as tweets match the criteria, they are pushed to the user. Tweepy - An easy-to-use Python library for accessing the Twitter API is used to authenticate and retrieve tweets from twitter. The retrieved data is then cleaned and only the tweets are fetched from it which is fed to a classifier. The classifier decides if the tweets are of positive, negative or neutral reaction. This data is then visualized according the user requirements.

Specifications



Software

Operating System	OS X, Windows, Ubuntu
Programming Language	Python 2.7 or above
Python Packages	Numpy, Pandas, Matplotlib, Tweepy, NLTK, TextBlob
Storage	Spreadsheet



Hardware

Processor	64 Bit Intel or AMD CPU
Primary Memory	4 Gigabytes or above
Secondary Memory	50 Gigabytes or above
Network Connection	2 Megabits per sec or above

Design

Flow Diagram:

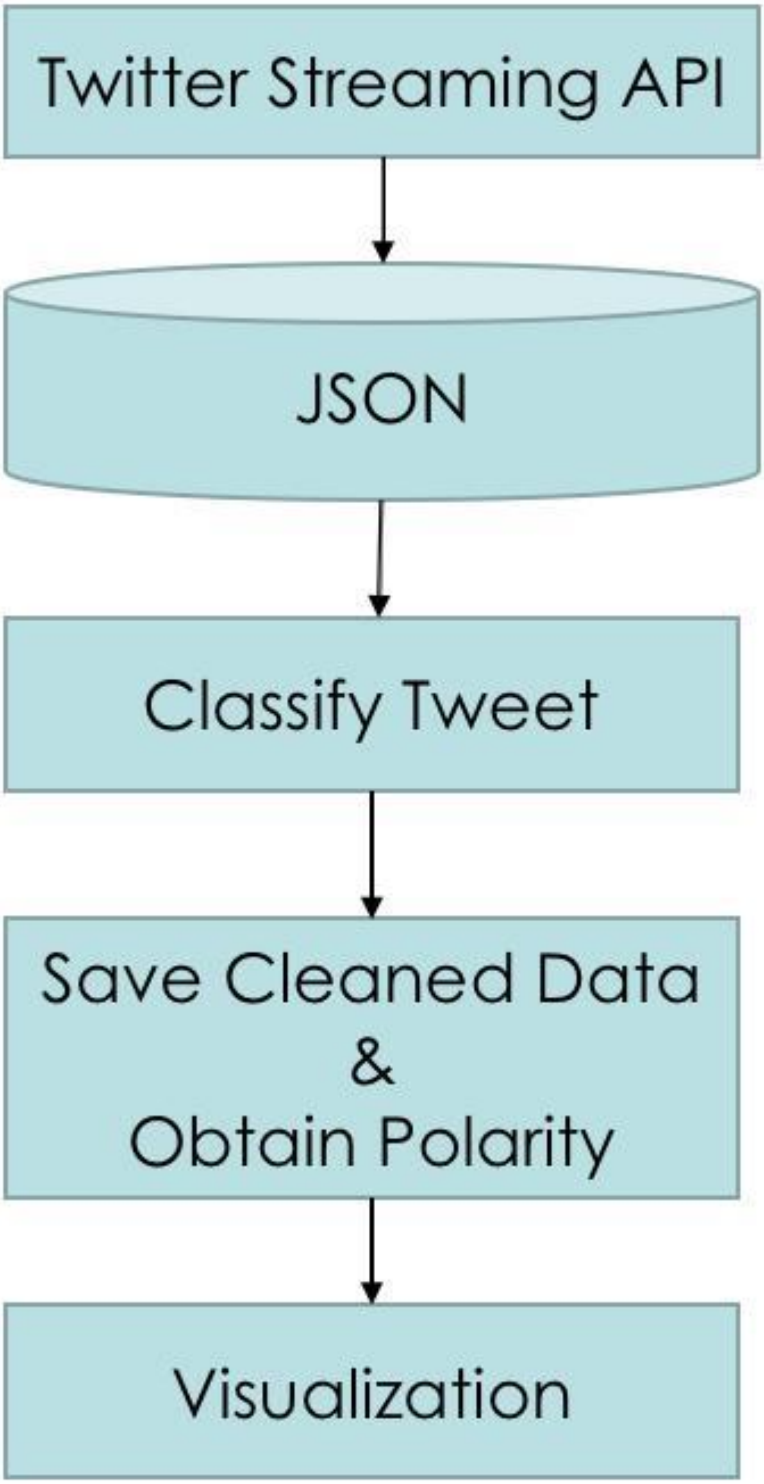


Fig 1

Architecture:

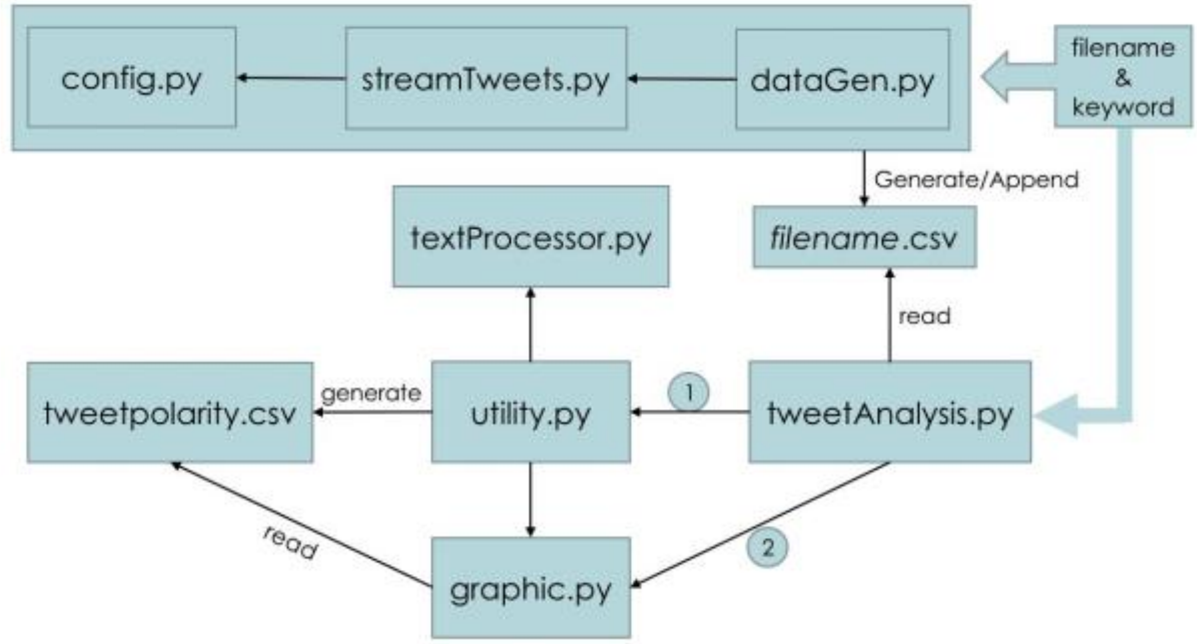


Fig 2

Modules:

TwitterStream.py: Function and class to gather tweets using twitter API and save in JSON format.

textProcessor.py: Functions to clean tweet and classify the tweet polarity using TextBlob.

Utility.py: Function to save cleaned data and polarity in a csv file, other functions.

graphic.py: Functions for different kinds of charts.

Result

A .csv file is generated which has the tweets, timestamps and the polarity of the tweets.

Various graphs are plotted using the data from this file which is used to infer the reaction of the people towards a particular event.

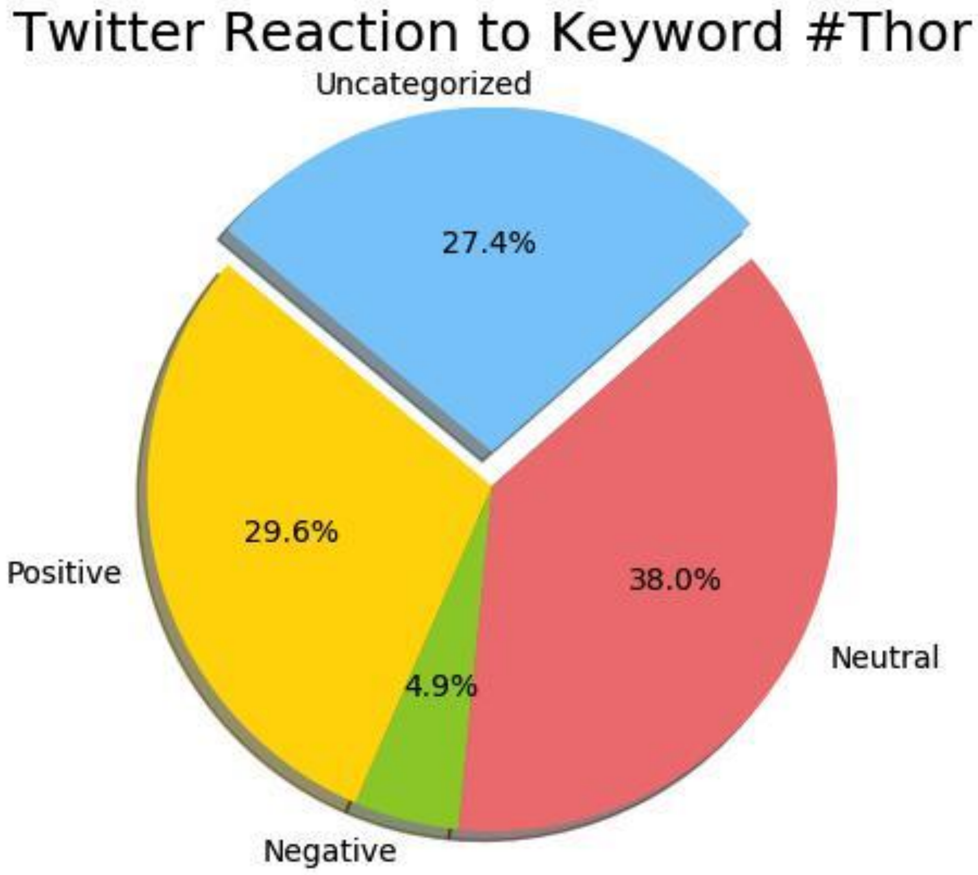


Fig 3

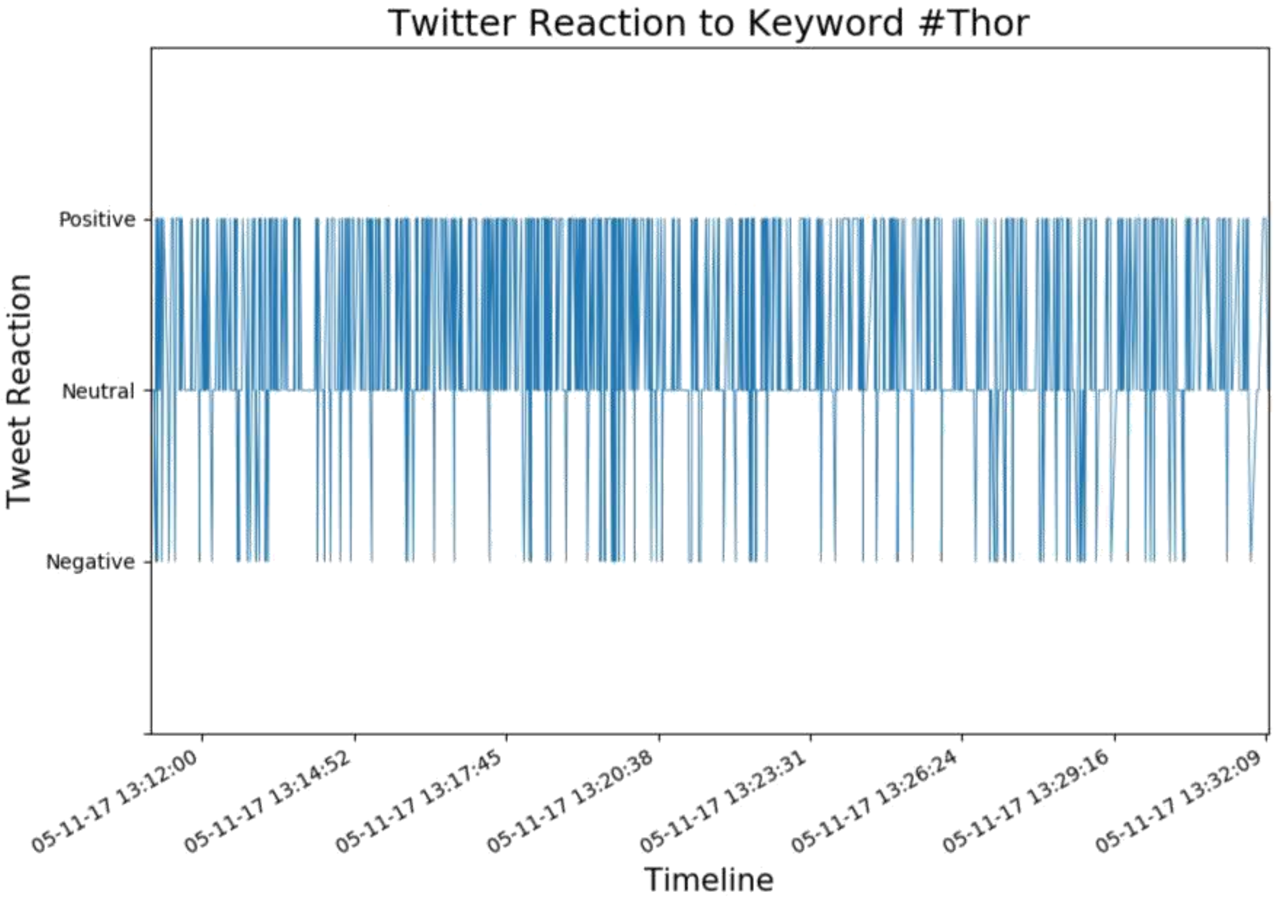

Twitter Reaction to Keyword #Thor
Tweet Reaction
Positive
Neutral
Negative
05-11-17 13:12:00
05-11-17 13:14:52
05-11-17 13:17:45
05-11-17 13:20:38
05-11-17 13:23:31
05-11-17 13:26:24
05-11-17 13:29:16
05-11-17 13:32:09
Timeline

Fig 4

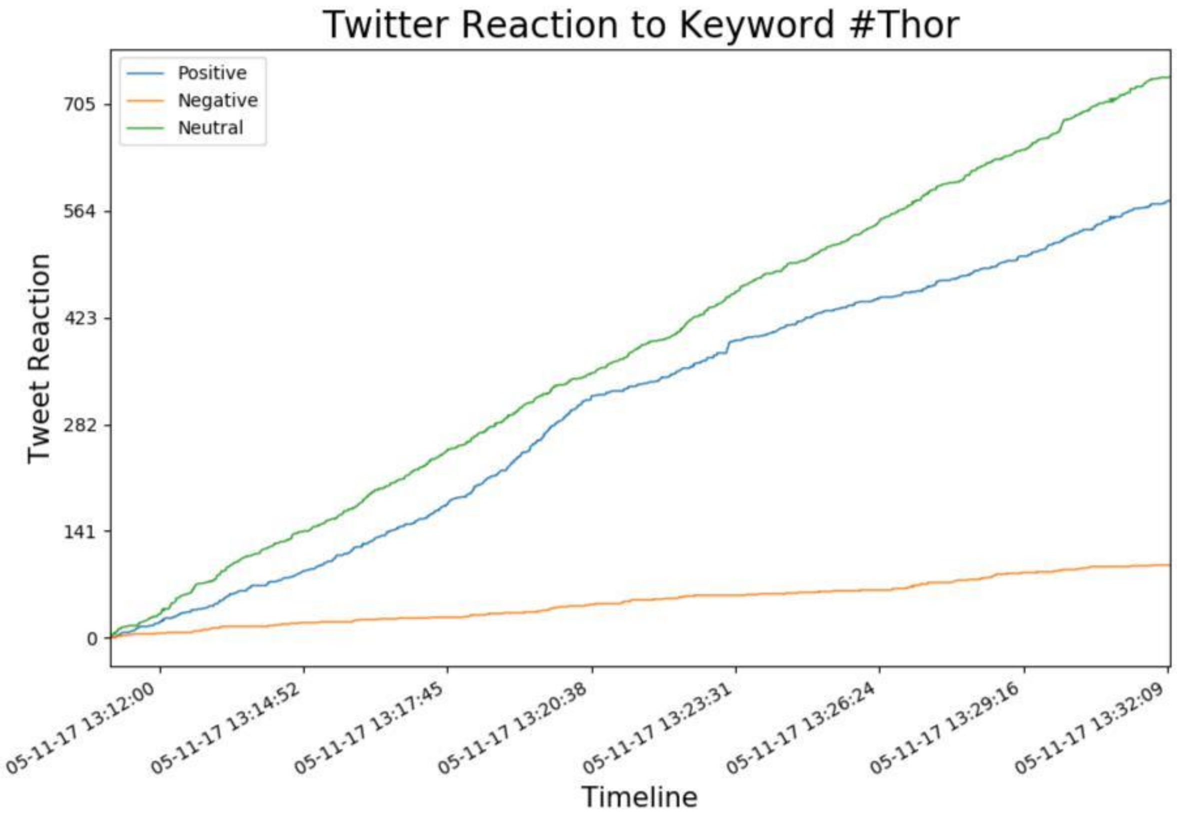

Twitter Reaction to Keyword #Thor
Positive
Negative
Neutral
Tweet Reaction
705
564
423
282
141
0
05-11-17 13:12:00
05-11-17 13:14:52
05-11-17 13:17:45
05-11-17 13:20:38
05-11-17 13:23:31
05-11-17 13:26:24
05-11-17 13:29:16
05-11-17 13:32:09
Timeline

Fig 5

Scope for further work

Implementing a design to do real time visualizations.

In the current model, data fetching from twitter is done in one phase and the processing and visualization is done in the second phase. This can be further improved to view visualizations as the data is being collected from twitter providing real time analysis.

Integrating online classifiers like IBM Watson, Google NLP to improve accuracy of classifications.

Individual Contribution

- Data Extraction,
  The data received using Streaming API was complex and had lot of redundant information. Only data according to the requirement like date, text, etc… were extracted from the complex chunk and processed further.

- Text Processing,
  Retrieved data is a complex JSON object which includes tweets as well. The tweet text is accompanied by hashtags, URLs, mentions, smileys, etc... which makes it difficult to classify further. These redundant data had to be cleaned using regular expression so that a proper clean tweet text can be obtained.

- Data Transformation,
  The time stamp provided from twitter is in epoch time which is not recognisable by python and matplotlib library. The time had to be converted to python format and then again to matplotlib date format so that it can be properly represented, which was a meticulous task.

- Data Visualization,
  Once the structured data is classified, it had to be visualized using various graphs to make further analytical inference. Various graphical charts like line chart and multiline chart had to be implemented to visualize data aesthetically.