IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727
PP 09-12
www.iosrjournals.org

A Survey on Opinion Mining Tools and Techniques for Tweets

Mrs.D.Suganthi[1], Dr.A.Geetha[2]
[1](*M.Phil Research Scholar, Department of Computer Science, Chikkanna Government Arts College, Tirupur, India)*
[2](*Assistant Professor, Department of Computer Science, Chikkanna Government Arts College, Tirupur, India)*

Abstract: *Opinion mining refers to computational techniques for analyzing the opinions that are extracted from various data sources. Opinion mining involves computational treatment of opinion and subjectivity in text. Before making any decision it is necessary to analyze what other people think. Customers or other people post their opinion, review, experience and feedback about various products, services and government schemes. The amount of data in twitter is massive so, other people and customer cannot read all reviews or opinions. Opinion mining is the appropriate technique to analyze different opinions of the customers or people. The various opinion mining tools and techniques are discussed in this paper.*

Keywords – *classification, feature extraction, micro blogging, sentiment analysis, tokenization*

I. Introduction

Opinion mining is also called sentiment analysis or opinion extraction is the field of study that analyses opinions, reviews, sentiments, feedbacks, experiences and suggestions about products, services, events / issues and government announced schemes. Twitter and 'tweeting' is about broadcasting daily short burst messages to the world, with the hope that someone's messages are useful and interesting to someone, it is known as *microblogging*. Every second, on average, around 6,000 tweets are tweeted on Twitter, which corresponds to over 350,000 tweets sent per minute, 500 million tweets per day and around 200 billion tweets per year. Every day, Twitter users generate tons of data; all those 140-character messages add up to 12 terabytes of data every day. This wealth of data seems overwhelming. Opinion mining is type of natural language processing that tracks people opinions and reviews about products or services or any topic.

Opinion mining tasks can be generally classified into three types. The first task is referred to as sentiment analysis and aims at the establishment of the polarity of the given source text (e.g., distinguishing between negative, neutral and positive opinions) [1]. The second task consists in identifying the degree of objectivity and subjectivity of a text (i.e., the identification of factual data as opposed to opinions). This task is sometimes referred to as opinion extraction. The third task is aims at the discovery and summarization of explicit opinions on selected features of the assessed product [2]. Fig 1. Depicts the tasks of opinion mining.

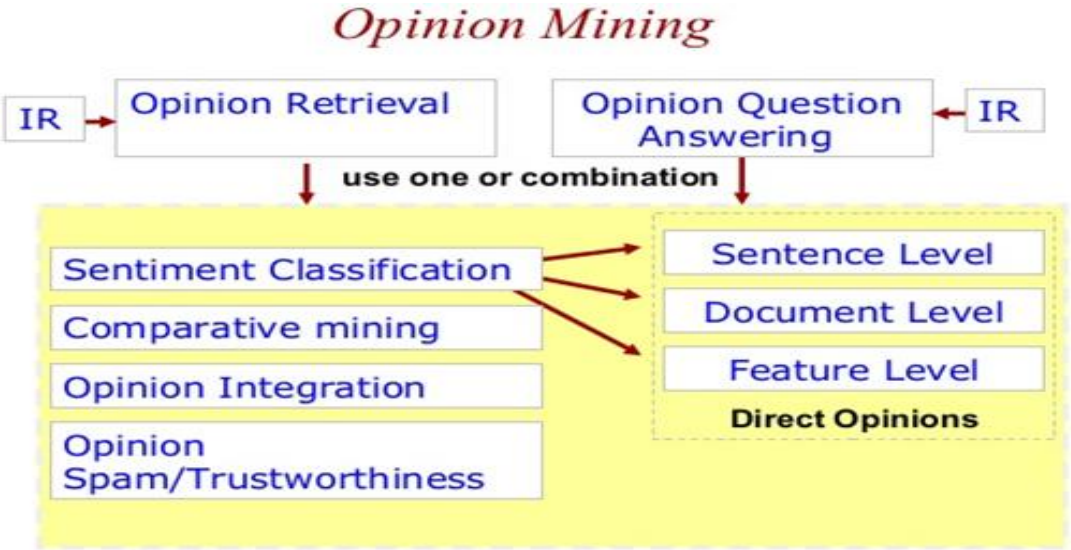

Fig: 1 Opinion Mining Tasks

II. Sentiment Analysis

1.1 Levels of analysis

1.1.1 Sentence level

The task at this level goes to the sentences and determines whether each sentence expressed a positive, neutral or negative opinion. The first step is to identify whether the sentence is subjective or objective.

1.1.2 Document level

The task at this level is to classify whether a whole opinion document expresses a positive or negative sentiment.

1.1.3 Feature level

Both the sentence and document level analysis do not discover what exactly people liked and did not like. Instead of looking at language constructs, aspect level directly looks at the opinion itself.

1.2 Comparisons

Identify comparative sentences & extract comparative relations.

a) Opinion integration - Automatically integrates opinions from different data sources.
b) Opinion spam / trustworthiness - To determine likelihood of spam in opinion and also determine authority of opinion
c) Opinion retrieval - Analogous to document retrieval process, requires documents to be retrieved and ranked according to opinions about topic.
d) Opinion question answering - Similar to opinion retrieval task, only that instead of returning a set of opinions, answers have to be a summary of those opinions and format has to be in natural language form.

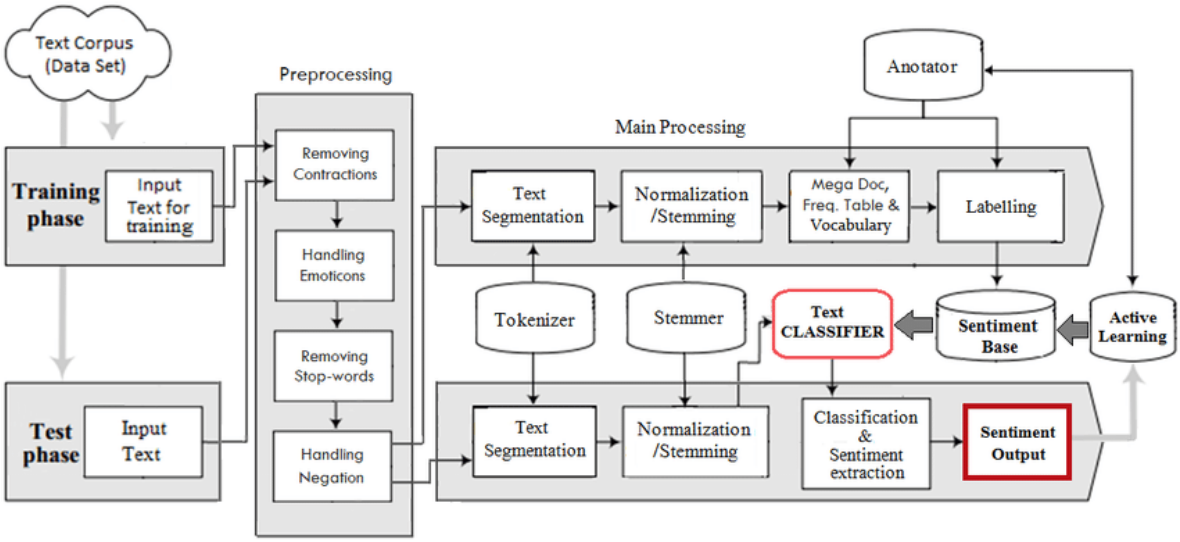

Fig 2. Opinion mining System architecture

III. Opinion Mining Techniques

3.1 Extraction of Tweets

3.1.1 Tools

Beautiful Soup — A useful Python library for scraping web pages that has extensive documentation and community support. Choosing elements to save from a page is as simple as writing a CSS selector [3].

Twitter API - A Python wrapper for performing API requests such as searching for users and downloading tweets. This library handles all of the OAuth and API queries and provides it in a simple Python interface. Be sure to create a Twitter App and get the OAuth keys to get access to Twitter's API.

MongoDB - An open source document storage database and is the go-to "NoSQL" database. It makes working with a database feel like working with Javascript.

PyMongo - A Python wrapper for interfacing with a MongoDB instance. This library lets connect to Python scripts with database and read/insert records.

Cronjobs - A time based job scheduler that lets run scripts at designated times or intervals (e.g. always at 12:01 a.m. or every 15 minutes).
Firehose - If we want to go beyond the data limits that Twitter imposes for free access, we can upgrade to Twitter's Firehose API where we can get nearly unlimited access to Twitter's data stream via one of the various data providers that Twitter partners with, including Dataminr (CNN recently partnered with Dataminr build an application that alerts journalists in newsrooms of breaking news and emerging trends), Datasift, Gnip, Lithium, Topsy.

3.2 Preprocessing

3.2.1 Tokenization

Tokenization is the process of breaking a stream of text into words, phrases, symbols or other meaningful elements called tokens .The aim of the tokenization is the exploration of the words in a sentence. The list of tokens becomes input for further processing such as parsing or text mining. Fig 2 shows that the system architecture of overall opinion mining.

a) Nlpdotnet Tokenizer: In computational linguistics, Nlpdotnet is a Python library for Natural Language Processing tasks which is based on neural networks. Currently, it performs tokenization, part-of-speech tagging, semantic role labelling and dependency parsing. Though it seems trivial, tokenizing is so important that it is vital to almost all advanced natural language processing activities.

b) Mila Tokenizer: MILA was developed in 2003 by the Israel Ministry for Science and Technology. Its mission is to create the infrastructure necessary for computational processing of the Hebrew language and make it available to the research community as well as to commercial enterprises. The MILA Hebrew Tokenization Tool divides inputted undotted Hebrew text (right to left) into tokens, sentences, and paragraphs and converts these into XML format.

c) NLTK Word Tokenize: NLTK stands for Natural Language Tool Kit. It is the famous Python Natural Language Processing Toolkit. NLTK is an important platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora (a corpus (plural corpora) is a large and structured set of texts) and lexical resources such as WordNet, along with a set of text processing libraries for classification, tokenization, stemming, tagging.

d) TextBlob Word Tokenize: TextBlob is a new python based natural language processing toolkit, which carries the fields like NLTK and Pattern. It provides text mining, text analysis and text processing modules for the python developers. It contains the python library for processing the textual form of data. It provides a simple application program interface (API) for leaping into common natural language processing (NLP) tasks, such as tokenizing, part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation and more.

e) MBSP Word Tokenize: MBSP is a text analysis system and it is based on TiMBL (Tilburg Memory-Based Learner) and MBT (Memory Based Tagger) memory based learning applications developed at CLiPS (Computational Linguistics & Psycholinguistics) and ILK (Induction of Linguistic Knowledge). It provides tools for Tokenization and Sentence Splitting, Part of Speech Tagging, Chunking, Lemmatization, Relation Finding and Prepositional Phrase Attachment. The general english version of MBSP has been trained on data from the Wall Street Journal corpus. The Python implementation of MBSP is open source and freely available.

f) Pattern Word Tokenize: Pattern is a web mining module for the Python programming language. It has tools for data mining (Google, Twitter and Wikipedia API, a web crawler, a HTML DOM parser), natural language processing (part-of-speech taggers, preprocessing, n-gram search, sentiment analysis, WordNet), machine learning (vector space model, clustering, SVM), network analysis and canvas visualization.

g) Word Tokenization with Python NLTK: NLTK provides a number of tokenizers in the module. The text is first tokenized into sentences using the PunktSentenceTokenizer. Then each sentence is tokenized into words using four different word tokenizers:

- TreebankWordTokenizer - This tokenizer uses regular expressions to tokenize text as in Treebank.
- WordPunctTokenizer - This tokenizer divides a string into substrings by splitting on the specified string, which it is defined in subclasses.

- PunctWordTokenizer- This tokenizer divides a text into a list of sentences; by using unsupervised algorithms.
- WhitespaceTokenize - This tokenizer divides text at whitespace.

3.2.2 Stop words removal

Many words in documents recur very frequently but are essentially meaningless as they are used to join words together in a sentence. It is commonly understood that stop words do not contribute to the context or content of textual documents.

Due to their high frequency of occurrence, their presence in text mining presents an obstacle in understanding the content of the documents. Stop words are very frequently used common words like 'and', 'are', 'this' etc. They are not useful in classification of documents. So they must be removed.

a) NLTK: NLTK (Natural Language Toolkit) in python has a list of stop-words stored in 16 different languages. You can find them in the nltk_data directory home/pratima/nltk_data/corpora/stopwords is the directory address.

3.3 Feature extraction

a) Unigram features – one word is considered at a time and decided whether it is capable of being a feature.
b) N-gram features – more than one word is considered at a time.
c) External lexicon – use of list of words with predefined positive or negative sentiment.

3.4 Classification algorithms

a) Naive Bayes: A probabilistic classifier that can learn the patterns by examining the set of documents performed is known as a Naïve Bayes classification process. The classification of documents according to their category or class is done with the help of comparison with the contents present within the words [4].

b) Maximum entropy: With the help of accessing conditional distribution of the class label, the entropy of the system is increased to the highest with the help of maximum entropy process. However, there are no assumptions made related to the relationship present within the features extracted from the dataset. The overlap feature is also handled within this process. This method is very similar to the logistic regression method in which various distributions over the classes are recognized [5].

c) Support vector machine: The analysis of data, characterization of the decision boundaries and the involvement of kernels within the computations that are done in the input space is done within the Support vector machine. At that point each data which represented as a vector is classified into a class. Facilitate one finds a margin between the two classes that is a long way from any document. The distance defines the margin of the classifier, amplifying the margin diminishes indecisive decisions. SVM additionally supports classification and regression which are valuable for statistical learning theory and it likewise helps perceiving the factors precisely, that should be considered, to comprehend it successfully.

IV. Conclusion

The main purpose of this paper is to present the various tools and techniques used in opinion mining or sentiment analysis. Here very familiar and open source tools are given such as different tokenizer. Some of the frequent classification techniques are presented. This paper will be very helpful to the researchers, seeking for opinion mining tools and techniques.

References

[1] G. Vinodhini and RM. Chandrashekharan, "Sentiment Analysis and Opinion Mining : A Survey" – International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 6, June 2012.
[2] Sajin. S. Chandran, Murugappan S., "A Review on Opinion Mining from Social Media Networks", European Journal of Scientific Research, pp.430-440, 3rd October, 2012.
[3] Ayesha Rashid1, Naveed Anwer2, Dr. Muddaser Iqbal3, Dr. Muhammad Sher4 "A Survey Paper: Areas, Techniques and Challenges of Opinion Mining", IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 6, No 2, November 2013.
[4] Kirti Huda, Md Tabrez Nafis, Neshat Karim Shaukat, "Classification Technique for Sentiment Analysis of Twitter Data" , International Journal of Advanced Research in Computer Science, Volume 8, No. 5, May-June 2017 ISSN No. 0976-5697
[5] M. Govindarajan, Romina M, " A Survey of Classification Methods and Applications for Sentiment Analysis" – International Journal of Engineering and Science (IJES), Volume 2, Issue 12, 2013.