IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 17, Issue 6, Ver. I (Nov – Dec. 2015), PP 157-162
www.iosrjournals.org

Sentiment of Sentence in Tweets: A Review

Sandip Mali[1], Ashish Balerao[1], Suvarnsing Bhable[1], Sangramsing Kayte[1,]
[1,] Department of Computer Science and Information Technology
Dr. Babasaheb Ambedkar Marathwada University, Aurangabad

Abstract: *Determine the sentiment of sentence that is positive or negative based on the presence of part of speech tag, the emoticons present in the sentences. For this research we use the most popular microblogging sit twitter for sentiment orientation. In this paper we want to extract tweets form the twitter related to the product like mobile phones, home appliances, vehicle etc. After retrieving tweets we perform some preprocessing on it like remove retweets, remove tweets containing few words with minimum threshold of length five, remove tweets containing only urls. After this the remaining tweets are pre-processed like that transform all letters of the tweets to the lower case then remove punctuation from the tweets because it reduces the accuracy of result. After this remove extra white spaces from the tweets, then we apply a pos tagger to tag each word. The tuple after the applying above steps contain (word, pos tag, English-word, stop-word). We are interested in only tweets that contain opinion and eliminate the remaining non-opinion tweets from the data set. For this we use the Naïve Bays classification algorithm. After this we use short text classification on tweets i.e., the word having different meaning in different domain. In order to solve this problem we use two different feature selection algorithms the mutual information (MI) and the X2 feature selection. At final stage predicting the orientation of an opinion sentence that is positive or negative as we mentioned above. For this we use two model like unigram model and opinion miner.*

Keywords: *Compositional Semantic Rule Algorithm, Numeric Sentiment Identification Algorithm, Bag-of-Word and Rule-based Algorithm, CRF Tagger, POS tagger*

I. Introduction

Twitter is a popular real-time microblogging service that allows its users to share short pieces of information known as “tweets”, means tweet is the small text that would be generated by user related to certain things like product, his own opinion, his beliefs etc. The only problem with tweet is that its length should be less than 140 characters. First we will introduce various properties of messages that users post on Twitter. Some of the many unique properties include the following:

a) Usernames: Users often include Twitter usernames in their tweets in order to direct their messages. A de facto standard is to include the @ symbol before the username (e.g @liang).
b) Hash Tags: Twitter allows users to tag their tweets with the help of a “hash tag”, which has the form of #<tagname>”. Users can use this to convey what their tweet is primarily about by using keywords that best represent the content of the tweet.
c) RT: If a tweet is compelling and interesting enough, users might republish that tweet, commonly known as retweeting, and twitter employs “RT” to represent re-tweeting (e.g. “RT @RodyRoderos: I love iphone 6 but i want Samsung note 2 :(”).

Tweets are also called as the microblog because of its short text. Microblogging websites have evolved to become a source of varied kind of information. This is due to nature of microblogs on which people post real time messages about their opinions on a variety of topics, discuss current issues, complain, and express their opinion for products they use in daily life. Due to this, Microblogging websites have evolved to become a source of a diverse variety information, with millions of messages appearing daily on popular web-sites. Product reviewing has been rapidly growing in recent years because more and more products are selling on the Web. The large number of reviews allows customers to make informed decisions on product purchases. However, it is difficult for product manufacturers or businesses to keep track of customer opinions and sentiments on their products and services. In order to enhance the customer shopping experiences a system is needed to help people analyze the sentiment content of product reviews.

A) Why opinions are important?

Opinions are central to almost all human activities because they are key influencers of our behaviors. Whenever we need to make a decision, we want to know others’ opinions. In the real world, businesses and organizations always want to find consumer or public opinions about their products and services. Individual consumers also want to know the opinions of existing users of a product before purchasing it, and others’ opinions about political candidates before making a voting decision in a political election. In the past, when an individual needed opinions, he/she asked friends and family. When an organization or a business needed public or consumer opinions, it conducted surveys, opinion polls, and focus groups. Acquiring public and consumer opinions has long been a huge business itself for marketing, public relations, and political campaign companies.

B) Sentiment analysis and Opinion mining

Most of the user-generated messages on microblogging websites are textual information, identifying their sentiments have become an important issue. The research in the field started with sentiment classification, which treated the problem as a text classification problem. Textual information in the world can be broadly classified into two main categories, facts and opinions. Facts are objective statements about entities and events in the world. Opinions are subjective statements that reflect people's sentiments or perceptions about the entities and events. For example "Delhi is capital of the India" is a factual type of information and "After watching movie Bahubali, I feel like there is no waste of time and money" is the subjective sentence indicates positive of opinion on film Bahubali in response with time and money. One important difference in facts and opinion is facts are same for all but different people have different opinions on the same thing. The term sentiment analysis and opinion mining basically represents the same field of study. Sentiment analysis, also called opinion mining, is the field of study that analyzes people's opinions, sentiments, evaluations, appraisals, attitudes, and emotions towards entities such as products, services, organizations, individuals, issues, events, topics, and their attributes.

It represents a large problem space. There are also many names and slightly different tasks, e.g., sentiment analysis, opinion mining, opinion extraction, sentiment mining, subjectivity analysis, affect analysis, emotion analysis, review mining, etc. However, they are now all under the umbrella of sentiment analysis or opinion mining. The term sentiment analysis perhaps first appeared in (Nasukawa and Yi, 2003), and the term opinion mining first ppeared in (Dave, Lawrence and Pennock, 2003). The term sentiment analysis and opinion mining is although coined in the linguistic and natural language processing, the little research had been done before 2000. But after it the researchers get focused on the sentiment analysis in the domain of natural language processing as major research area, because it has wide range of commercial application almost in every domain.

C) Different levels of Sentiment Analysis –

In general, sentiment analysis has been investigated mainly at three levels: Document level Sentence level Entity and aspect level Document Level The task at this level is to classify whether a whole opinion document xpresses a positive or negative sentiment. For example, given a product review, the system determines whether the review expresses an overall positive or negative opinion about the product. This task is commonly known as document-level sentiment classification. This level of analysis assumes that each document expresses opinions on a single entity (e.g., a single product). Thus, it is not applicable to documents which evaluate or compare multiple entities. The recent research has shown that the even in a negative document there is more than 40% is positive text in it. Sentence Level The task at this level goes to the sentences and determines whether each sentence expressed a positive, negative, or neutral opinion. Neutral usually means no opinion. This level of analysis is closely related to subjectivity classification, which distinguishes sentences that express factual information from sentences that express subjective views and opinions. The document is nothing but the collection of the sentences together but the accuracy of sentence level sentiment analysis is much fine than the document level.

Entity and Aspect level It give much better result than both document level and sentence level sentiment analysis. Both the document level and the sentence level analyses do not discover what exactly people liked and did not like. Aspect level performs finer-grained analysis. Instead of looking at language constructs (documents, paragraphs, sentences, clauses or phrases), aspect level directly looks at the opinion itself. It is based on the idea that an opinion consists of a sentiment (positive or negative) and a target (of opinion).

In this research our main aim is the findings the tweets that contain opinion and based on them later determine their orientation that is the tweet is contain either positive or negative or neutral polarity. As the users of microblogging platforms and services grow every day, data from these sources can be used in opinion mining and sentiment analysis tasks. For example, manufacturing / commercial companies may be interested in the following questions:

- What do people think about our product (service, company etc.)?
- How positive (or negative) are people about our product?
- What would people prefer our product to be like?

II. Literature Review

PAPER NO: 1 in this paper they propose a new system architecture that can be automatically analyze the sentiment of microblogs or tweets. They combine this system with manually annotated data from twitter which is one of the most popular microblogging platforms for the task of sentiment analysis. In this system, machines can learn how to automatically extract the set of messages which contain opinions, filter out non-opinion messages and determine their sentiment directions. For this paper, they crawl tweets from twitter and perform some preprocessing on it. They retrieve tweets using twitter API. They crawl tweets of three distinct categories (camera, mobile phone, movies) as their training set from the time period between November 1, 2012 to January 31, 2013. They perform some preprocessing task on that like eliminate tweets that are not in English, have too few words, have too few words apart from greeting words, have just URL. After all the remaining tweets are pre-processed as all words are transformed to lower case, extract emoticons with their sentiment polarity, targets are replaced with user, pos tagging, remove sequence of repeated characters and stop-words. According to the previous preprocessing step all words are transformed into a tupleof structure (word, pos tag, English-word, stop-word). In the next stage filter-out tweets without opinion, to do this they use Naive Bays (NB). In this step, the system can classify the tweets into opinion and non-opinion class. Then the system passes the opinion part into the next step i.e. short text classification. In this part they observed that a word may have different meaning s in different domains. For this they use two different algorithms like Mutual Information (MI), and X2 test. The final step of their work is to determine the orientation of the tweets i.e., positive or negative. In this paper they got accuracy about 67.58% for unigram and 70.39% for opinion miner. This result show that opinion miner give better result than unigram model.

PAPER NO: 2 in this paper, they look at one such popular microblogs called Twitter and build models for classifying "tweets" into positive, negative and neutral sentiment. They build models for two classification tasks: a binary task of classifying sentiment into positive and negative classes and a 3-way task of classifying sentiment into positive, negative and neutral classes. We experiment with three types of models: unigram model, a feature based model and a tree kernel based model. There experiments show that a unigram model is indeed a hard baseline achieving over 20% over the chance baseline for both classification tasks. There feature based model that uses only 100 features achieves similar accuracy as the unigram model that uses over 10,000 features. There tree kernel based model outperforms both these models by a significant margin. They also experiment with a combination of models: combining unigrams with our features and combining our features with the tree kernel. Both these combinations outperform the unigram baseline by over 4% for both classification tasks. They use manually annotated Twitter data for their experiments. One advantage of this data, over previously used data-sets, is that the tweets are collected in a streaming fashion and therefore represent a true sample of actual tweets in terms of language use and content. They acquire 11,875 manually annotated Twitter data from a commercial source. Each tweet is labeled by a human annotator as positive, negative, neutral or junk. They eliminate the tweets with junk. In this paper, they prepare the emoticon dictionary by labeling 170 emoticons and an acronym dictionary. They pre-process all the tweets as follows: a) replace all the emoticons with a their sentiment polarity, b) replace all URLs with a tag ||U||, c) replace targets with tag ||T||, d) replace all negations by tag "NOT", and e) replace a sequence of repeated characters by three characters. For all their experiments they use Support Vector Machines (SVM) and report averaged 5-fold cross-validation test results.

PAPER NO: 3 We introduce a novel approach for automatically classifying the sentiment of Twitter messages. These messages are classified as either positive or negative with respect to a query term. Their training data consists of Twitter messages with emoticons, which are used as noisy labels. They show that machine learning algorithms (Naive Bayes, Maximum Entropy, and SVM) have accuracy above 80% when trained with emoticon data. This paper also describes the preprocessing steps needed in order to achieve high accuracy. They strip the emoticons out from our training data, because there is a negative impact on the accuracy of the Max.Ent and SVM classifiers, but little effect on Naive Bayes. Stripping out the emoticons causes the classifier to learn from the other features (e.g. unigrams and bigrams) present in the tweet. They reduce the feature space by removing user with tag USERNAME, links with URL, and replacing the repeated sequence of the characters. They test different classifiers: keyword-based, Naive Bayes, Maximum entropy and support vector machines.

PAPER NO: 4 in this paper, they focus on using Twitter, the most popular microblogging platform, for the task of sentiment analysis. They show how to automatically collect a corpus for sentiment analysis and opinion mining purposes. They perform linguistic analysis of the collected corpus and explain discovered phenomena.

Using the corpus, they build a sentiment classifier that is able to determine positive, negative and neutral sentiments for a document. They collected a corpus of 300000 text posts from Twitter evenly split automatically between three sets of texts i.e., positive emotions, negative emotions and no emotions. They perform statistical linguistic analysis of the collected corpus. They use the collected corpora to build a sentiment classification system for microblogging. Using Twitter API they collected a corpus of text posts and formed a dataset of three classes: positive sentiments, negative sentiments, and a set of objective texts. They first checked the distribution of words frequencies in the corpus using Zipf's law; next, they used TreeTagger for English to tag all the posts in the corpus. They are interested in a difference of tags distributions between sets of texts.

They can observe that objective texts tend to contain more common and proper nouns (NPS, NP, NNS), while subjective texts use more often personal pronouns (PP, PP$). The collected dataset is used to extract features that will be used to train sentiment classifier. They used the presence of an n-gram as a binary feature. They build a sentiment classifier using the multinomial Naive Bayes classifier, SVM and CRF however the Naive Bayes classifier yields the best results. To increase the accuracy of the classification, they discard common n-grams, i.e. n-grams that do not strongly indicate any sentiment nor indicate objectivity of a sentence. They examined two strategies of filtering out the common n-grams: salience and entropy, the salience provides a better accuracy than entropy, therefore the salience discriminates common n-grams better then the entropy.

PAPER NO: 5 in this paper, they examine the effectiveness of applying machine learning techniques to the sentiment classification problem. They consider the problem of classifying documents not by topic, but by overall sentiment, e.g., determining whether a review is positive or negative. For their experiments, they chose to work with movie reviews. This domain is experimentally convenient because there are large on-line collections of such reviews, and because reviewers often summarize their overall sentiment with a machine-extractable rating indicator. Their data source was the Internet Movie Database (IMDb) archive of the" www.rec.arts.movies.reviews" newsgroup. This dataset is available on-line at http://www.cs.cornell.edu/people/pabo/-movie-review-data/. Their aim in this work was to examine whether it suffices to treat sentiment classification simply as a special case of topic-based categorization. They experimented with three standard algorithms: Naive Bayes classification, maximum entropy classification, and support vector machines In terms of relative performance, Naive Bayes tends to do the worst and SVMs tend to do the best, although the difference are not large.

PAPER NO: 6 This paper presents a simple unsupervised learning algorithm for classifying reviews as recommended (thumbs up) or not recommended (thumbs down). The classification of a review is predicted by the average semantic orientation of the phrases in the review that contain adjectives or adverbs. A phrase has a positive semantic orientation when it has good associations (e.g., "subtle nuances") and a negative semantic orientation when it has bad associations (e.g., "very cavalier"). In this paper, they present a simple unsupervised learning algorithm for classifying a review as recommended or not recommended. The algorithm takes a written review as input and produces a classification as output. The first step is to use a part-of-speech tagger to identify phrases in the input text that contain adjectives or adverbs. The second step is to estimate the semantic orientation of each extracted phrase. A phrase has a positive semantic orientation when it has good associations (e.g., "romantic ambience") and a negative semantic orientation when it has bad associations (e.g., "horrific events"). The third step is to assign the given review to a class, recommended or not recommended, based on the average semantic orientation of the phrases extracted from the review. If the average is positive, the prediction is that the review recommends the item it discusses. Otherwise, the prediction is that the item is not recommended. The PMI-IR algorithm is employed to estimate the semantic orientation of a phrase. PMI-IR uses Pointwise Mutual Information (PMI) and Information Retrieval (IR) to measure the similarity of pairs of words or phrases. The semantic orientation of a given phrase is calculated by comparing its similarity to a positive reference word ("excellent") with its similarity to a negative reference word ("poor"). In experiments with 410 reviews from opinions, the algorithm attains an average accuracy of 74%.

PAPER NO: 7 In this paper they present a system that, given a topic, automatically finds the people who hold opinions about that topic and the sentiment of each opinion. The system contains a module for determining word sentiment and another for combining sentiments within a sentence. They approach the problem in stages, starting with words and moving on to sentences. They take as unit sentiment carrier a single word, and first classify each adjective, verb, and noun by its sentiment. For word sentiment classification, the basic approach is to assemble a small amount of seed words by hand, sorted by polarity into two lists—positive and negative—and then to grow this by adding words obtained from WordNet. They are interested in the sentiments of the Holder about the Claim. They used BBN's named entity tagger IdentiFinder to identify potential holders of an opinion. They considered PERSON and ORGANIZATION as the only possible opinion holders. They built three models to assign a sentiment category to a given sentence, each combining the individual sentiments of sentiment-bearing words. Model 0 works something like "negatives cancel one another out". The Model 1 works something like this the number of words in the region whose sentiment category is c. If a region contains more and stronger positive than negative words, the sentiment will be positive. Model 2 works something like this the number of words in the region whose sentiment category is c. If a region contains more and stronger negative words than positive words, the sentiment will be negative. The best overall performance is provided by Model 0.

PAPER NO: 8 in this paper, they develop a sentiment identification system called SES which implements three different sentiment identification algorithms. They augment basic compositional semantic rules in the first algorithm. In the second algorithm, they think sentiment should not be simply classified as positive, negative, and objective but a continuous score to reflect sentiment degree. All word scores are calculated based on a large volume of customer reviews. Due to the special characteristics of social media texts, they propose a third algorithm which takes emoticons, negation word position, and domain-specific words into account. They build a web-based system called SES, which ensembles three algorithms we implemented and uses machine learning method to predict text sentiment. The system is aiming to predict sentiment on both sentence and document level. They conduct experiments on Facebook comments and twitter tweets using four different machine learning models: decision tree, neural network, logistic regression, and random forest. The experiment results show that random forest model reaches highest accuracy. The system contains multiple tasks from the input raw data towards output the final sentiment. The first task is data preprocessing. The second task is running three algorithms for each sentence to get three individual outputs. Then they generate features based on these outputs and put them into trained model to output sentiment. The fourth task is the combination task which combines all sentence sentiments to generate a overall sentiment of the input document. In this system, CRF Tagger, a java-based conditional random field part-of- speech (POS) tagger for English is employed to label each word. They have used different three algorithms for sentiment on a single sentence 1) Compositional Semantic Rule Algorithm, 2) Numeric Sentiment Identification Algorithm and 3) Bag-of-Word and Rule-based Algorithm.

PAPER NO: 9 Sentilo is an unsupervised, domain-independent system that performs sentiment analysis by hybridizing natural language processing techniques and semantic Web technologies. Sentilo combines natural language processing techniques with knowledge representation and makes use of affective knowledge resources such as SenticNet, SentiWord-Net and the SentiloNet resource of annotated verbs, presented as novel contribution in this article. Given a sentence expressing an opinion, Sentilo recognizes its holder, detects the topics and subtopics that it targets, links them to relevant situations and events referred to by it and evaluates the sentiment expressed on each topic/subtopic. Sentilo is a sentic computing approach to SA. It provides a formal representation of opinion sentences, in the form of RDF graphs. Sentilo approach to SA is based on the neo-Davidsonian assumption that events and situations are to be considered first class entities in the description of the world. As Sentilo represents sentences with an event-/situationbased approach, i.e., frame-based, it can benefit from the semantics of the frame-based relations between topics and subtopics for correctly and more deeply analyzing the sentiment of opinion sentences.

III. Data Set

The data used in this research is crawl by the R statistical tool of twitter tweets of the product like mobile phone etc by creating an twitter api. The tweets are crawl after the month January 2015 and the language of the tweet is only English. The tweets in other language are stripped from data set. After extracting tweets we create a corpus of tweets and use this corpus as input file for the next steps like preprocessing, sentiment orientation of tweets.

IV. Conclusion and Future scope

As we seen in the literature review the opinion miner and random forest algorithm gives the better result as compare with other algorithm. The main task that increase the accuracy is eliminates the tweets that do not contain strong opinion. This reduce the data set which we further use in later phase and thus increase overall speed of processing.

Further, a development in the feature extraction from the text is needed to increase the accuracy. The future scope is use the different datasets and applies various classification algorithms on it and checks the accuracy and try to increase it.

References

[1] Po-Wei Liang and Bi-Ru Dai. Opinion Mining on Social Media Data, DOI 10.1109/MDM. 2013.73, 978-0-7695-4973-6/13 © 2013 IEEE.

[2] Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambow, and Rebecca Passonneau. 2011. Sentiment analysis of twitter data. In Proceedings of the Workshop on Languages in Social Media, pages 30–38. Association for Computational Linguistics.

[3] Go, A., Huang, L., Bhayani, R.: Twitter sentiment classification using distant supervision. In: CS224N Project Report, Stanford (2009).

[4] Alexander Pak and Patrick Paroubek. 2010. Twitter as a corpus for sentiment analysis and opinion mining. Proceedings of LREC.

[5] 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.

[6] P. Turney. Thumbs Up or Thumbs Down? Semantic Orientation Appliedto Unsupervised Classification of Reviews. ACL'02, 2002.

[7] S. Kim and E. Hovy. Determining the Sentiment of Opinions.COLING'04, 2004.

[8] Kunpeng Zhang, Yu Cheng, Yusheng Xie, Daniel Honbo, Ankit Agrawal, Diana Palsetia, Kathy Lee , Wei-keng Liao , Alok Choudhary, SES: Sentiment Elicitation System for Social Media Data, Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops, p.129-136, December 11-11, 2011.
[9] Diego Reforgiato Recupero, Valentina Presutti, Sergio Consoli, Aldo Gangemi, Andrea Giovanni Nuzzolese. Sentilo: Frame-Based Sentiment Analysis, Received: 4 March 2014/ Accepted: 12 August 2014 / Published online: 2 September 2014 Springer Science + Business Media New York 2014.
[10] Bing Liu, Sentiment Analysis and Opinion Mining, April 22, 2012.