JOURNAL OF INFORMATION, KNOWLEDGE AND RESEARCH IN COMPUTER ENGINEERING (ISSN 0975 – 6760, GIF: 01.1135, SIF: 02.502)

Sentiment Analysis in Web Comments Using C++

Chaitra Bhat Hasanagi[#1], Harshitha M K[#2], K Soujanya[#3], Keerthi N[#4]

Department of Computer Science and Engineering
GSSS Institute of Engineering and Technology for Women, Mysuru

[1]chaitrabhat.13@gmail.com,[2]harshithamk29@gmail.com,[3]soujanya.k94@gmail.com[4]keerthishalinin@gmail.com

***Abstract*— Sentiment is a view or opinion expressed over something. Sentiment analysis or opinion mining is formally defined as the computational study of sentiments and opinions about an entity expressed in a text. There are multiple granularity levels of sentiment analysis, feature-level, entity-level, sentence-level, document-level.** [1] **In this work we consider sentiment analysis at sentence level. The goal of our sentiment analysis system is to obtain an output value that represents how much positive, negative or neutral is the sentiment expressed in the sentence.**

Keywords— sentiment analysis, polarity, modules, social network, web comments.

I. INTRODUCTION

Sentiment analysis refers to the use of natural language processing, text analysis and computational linguistics to identify and extract subjective information in source material [2]. It is the processing of computationally identifying and classifying the opinion expressed over a text in order to determine that opinion or attitude expressed as positive, negative or neutral. The basic task in sentiment analysis is classifying the polarity of a given text at the document or sentence.

In this paper we propose to perform sentiment analysis on the domain of social network because of challenging nature of category of texts. The rise in social media such as blogs and social networks has fueled interest in sentiment analysis. Reviews, ratings, recommendations & other form of online expression, online opinions has turned into a kind of virtual currency for business, looking to market their products, identifying new opportunities and manage their reputation.

Sentiment analysis is extremely useful in social network monitoring as it allows us to gain an overview of the wider public opinion behind certain topics [3]. We analyze the user generated text on the social network and classify the comments as being positive, negative or neutral.

This can be done in the following steps:

- Using C++ we build arrays to store list of good words and bad words.
- Extract the web comments and perform word by word analysis with the stored list of words.
- Categorize the analyzed comment as being positive, negative or neutral.
- Calculate the percentage of positivity or negativity i.e. the polarity.
- Determine the attitude of the speaker or a writer with respect to the topic.

Sentiment analysis example: [4]

(1) I've just registered for the competition.

(2) This movie is amazing.

(3) The reports are very good.

(4) Their blog is so informative.

(5)It is very expensive.

- •Sentence 1 is neutral, as it doesn't offer any sentiment.
- •Sentence 2 expresses a positive opinion about the movie as a whole.
- •Sentence 3 expresses a positive opinion about the movie's reports.
- •Sentence 4 expresses a positive opinion about the company's blog.
- •Sentence 5 expresses a negative opinion about a product's pricing.

In our work we perform sentiment analysis at sentence level.

Sentence level: examining the sentiment expressed in sentences.

Out of the 5 sentences, the first one doesn't express any sentiment, as it only states a fact. The remaining sentences express sentiment: the second, third, and

fourth sentences are positive, while the fifth one is negative.

II. RELATED WORK

This section presents different existing algorithms and works carried on sentiment analysis which gives a new motivation to enhance the working of our algorithm.

There have been many researches on the subject of sentiment analysis. There already exist sentiment analysis studies for movie/product reviews, blogs, and twitter messages.

The main research on sentiment analysis so far has mainly focused on two things:

-Identifying whether a given textual entity is subjective or objective

-And identifying the polarity of that subjective textual entity after removing the objective content.

Most sentiment analysis research is done with help of supervised machine learning techniques, but of course there are also some researches that use unsupervised machine learning techniques and/or statistical approaches. For machine learning approaches, a bag of words representation is used. Recently, there have been feature based approaches to improve results.

1) Sentiment Analysis Using Social Multimedia [5]

Sentiment analysis is one of the most active research areas in natural language processing, web/social network mining, and text/multimedia data mining. The growing importance of sentiment analysis coincides with the popularity of social network platforms, such as Facebook, Twitter, and Flickr, which provide a rich repository of people's opinion and sentiment about a vast spectrum of topics. Moreover, the fact that we are exposed to a tremendous amount of data in different forms including text, images, and videos makes sentiment analysis a very challenging task due to its nature of multimodality. This paper discusses some of the latest works on topics of sentiment analysis based on visual content and textual content.

2) Document Sentiment Classification [6]

2.1) Sentiment Classification Using Supervised Learning:

For the text classification problem, any existing supervised learning method can be applied, e.g., naïve Bayes classification, and support vector machines (SVM) (Joachims, 1999; Shawe-Taylor and Cristianini, 2000). Pang, Lee and Vaithyanathan (2002) was the first paper to take this approach to classify movie reviews into two classes, positive and negative. It was shown that using unigrams (a bag of words) as features in classification performed quite well with either naïve Bayes or SVM, although the authors also tried a number of other feature options.

2.2) Sentiment Classification Using Unsupervised Learning:

Since sentiment words are often the dominating factor for sentiment classification, it is not hard to imagine that sentiment words and phrases may be used for sentiment classification in an unsupervised manner. It performs classification based on some fixed syntactic patterns that are likely to be used to express opinions. The syntactic patterns are composed based on part-of-speech (POS) tags.

2.3) Cross-Domain Sentiment Classification:

Domain adaptation or transfer learning is needed. Existing researches are mainly based on two settings. The first setting needs a small amount of labeled training data for the new domain (Aue and Gamon, 2005). The second needs no labeled data for the new domain (Blitzer, Dredze and Pereira, 2007; Tan et al., 2007). The original domain with labeled training data is often called the source domain, and the new domain which is used for testing is called the target domain.

2.4) Cross-Language Sentiment Classification:

Cross-language sentiment classification means to perform sentiment classification of opinion documents in multiple languages.

There are many established methods for sentiment analysis at the sentence and paragraph level. [7]

-Mullen and Collier 2004 discussed the application of support vector machines in sentiment CS 144 Ideas Behind the Web Team: Dai Wei, Doris Xin analysis with diverse information source.

-Bang and Lee 2004 applied minimum cuts in graphs to extract the subjective portion of texts they were studying and used machine learning methods to perform sentiment analysis on those snippets of texts only.

-Wilson et al 2005 discussed categorizing texts into polar and neutral first before determining whether a positive or negative sentiment is expressed through the text.

The followings are a selection of works directly related to the project:

- Automatic Sentiment Analysis in On-line Text: [8]

This work provides a good survey of various techniques developed in online sentiment analysis. It covers concept of emotion in written text (appraisal theory), various methodologies which can be broadly divided into two groups: (i) symbolic techniques that focuses on the force and direction of individual words (the so-called "bag-ofwords" approach), and (ii)

machine learning techniques that characterizes vocabularies in context. Based on the survey, Boisy et al found that symbolic techniques achieves accuracy lower than 80% and are generally poorer than machine learning methods on movie review sentiment analysis. Among the machine learning methods, they considered three supervised approaches: support vector machine (SVM), naive Bayes multinomial (NBM), and maximum entropy (Maxent). They found that all of them deliver comparable results on various feature extraction (unigrams, bigrams, etc) with high accuracy at 80%~87%.

- Large-scale sentiment analysis for news and blogs: [9]

N. Godbole, M. Srinivasaiah, and S. Skiena developed techniques that algorithmically identify large number (hundreds) of adjectives, each with an assigned score of polarity, from around a dozen of seed adjectives. Their methods expand two clusters of adjectives (positive and negative word groups) by recursively querying the synonyms and antonyms from WordNet. Since recursive search quickly connects words from the two clusters, they implemented several precaution measures such as assigning weights which decrease exponentially as the number of hops increases. They confirm that the algorithm-generated adjectives are highly accurate by comparing them to the results of manually picked word lists. It is worth pointing out that this work uses Lydia as the backbone to process large amount of news and blogs.

- Twitter as a Corpus for Sentiment Analysis and Opinion: [10]

Pak et al took a naive approach to collect and classify 300000 tweets into three categories: (i) tweets queried with emoticon queries such as ":-)", ":)", "=)" indicate happiness and positive emotion (ii) tweets with ":-(", ":(", "=(", ";(" implies dislike or negative opinions, and (iii) tweets posted by newspaper accounts such as "New York Times" are considered objective or neutral. This serves as the training set for naive Bayes multinomial (NBM), which they found to be superior to SVM and CRF (Lafferty et al., 2001) as the classifier to unigrams, binagrams, and trigrams. The result indicates that bigrams provides the best accuracy.

- Outtweeting the Twitterers: Predicting Information Cascades in Microblogs: [11]

Instead of focusing on the natural language in tweets, Gluba et al tracks 15 million URL exchanged among 2.7 million Twitter users. Their data analysis in the cascading of URL uncovers social graphs and other properties on Twitter network. They further formulate a model to predict URL cascading, which accounts for more than half of the URL spread with low false-positive rate.

III. PROPOSED ALGORITHM

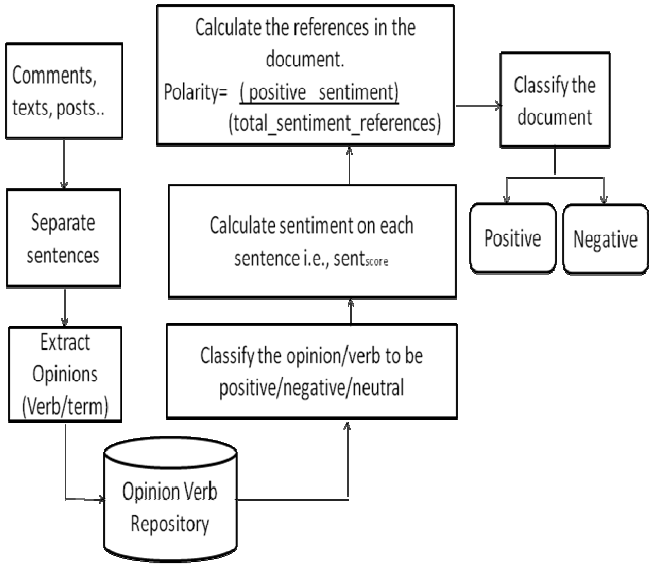
This section presents how we perform sentiment analysis on the web comments using C++. The previous section shows various existing algorithms and methods to perform sentiment analysis using JAVA. This proposed algorithm aims to achieve the performance of those existing algorithms using C++.

The Proposed algorithm contains following features:

- Function to extract web comments.
- Function to divide and categorize the text into positive or negative.
- Function that matches words (from comments) to the built list of words.
- Function to calculate the polarity of statements.

A. *Architecture:*

The system architecture is as shown below:



B. *Modules:*

This algorithm can be divided into 5 modules:

1) Creation of vocabulary and data sets:

- Build two separate lists namely
  - Positive/Good words.
    Ex: good, awesome, useful, pretty
  - Negative/Bad words.
    Ex: rude, dirty, worst, horrible

2) Input:

Extract comments from web pages.

3) Matching:

- *After extracting the context*

-divide it into sentence
-perform word-by-word analysis of each sentence.

-categorize each word to be positive, negative or neutral.

4) Calculate Polarity:

- Polarity indicates the percentage of positive sentiment references among total sentiment references.

$$\text{Polarity} = \frac{(\text{positive\_sentiment})}{(\text{total\_sentiment\_references})}$$

5) Output:

Display the calculated polarity to the end user.

Ex: Total words, Positive and Negative words, Polarity.

C. Algorithm:

The proposed algorithm is as below:

Step1: Define the goodWords[ ] and badWords[ ] arrays

Step2: Extract/Input the comment from user i.e. text[ ]

Step3: Divide each sentence in the text to form array of words i.e. input[ ]

Step4: For each word in the string input[ ], check whether it is present in the list of goodWords[ ] and badWords[ ] arrays

i. If the word is found in goodWords[ ] list, increment the count of positive_sentiment and also the total_sentiment_references

ii. If the word is found in badWords[ ] list, increment the count of negative_sentiment and also the total_sentiment_references

Step5: Calculate the polarity of the given text

$$\text{Polarity} = \frac{(\text{positive\_sentiment})}{(\text{total\_sentiment\_references})}$$

Step6: Display the calculated polarity to the end user.

I.e. Total words, Positive and negative words, Polarity.

IV. CONCLUSION

Sentiment analysis is an emerging research area in text mining and computational linguistics, and has attracted considerable research attention in the past few years. Our proposed algorithm performs sentiment analysis using C++. Future research shall explore sophisticated methods for opinion and product feature extraction, as well as new classification models.

References

[1] Davide Feltoni Gurini, Fabio Gasparetti, Alessandro Micarelli and Giuseppe Sansonetti, "A Sentiment-Based Approach to Twitter User Recommendation", http://ceur-ws.org/Vol-1066/Paper5.pdf

[2] https://en.wikipedia.org/wiki/Sentiment_analysis

[3] https://www.brandwatch.com/2015/01/understanding-sentiment-analysis/

[4] Ben Donkor, "On Social Sentiment and Sentiment Analysis", December 16,2013.

[5] Jianbo Yuan, Quanzeng You and Jiebo Luo, "Sentiment Analysis Using Social Multimedia", chapter 2.

[6] Bing Liu, "Sentiment Analysis and Opinion Mining", April 22, 2012, pp 30-43

[7] Dai Wei, Doris Xin, "Project Plan: Sentiment Analysis on Tweets and Facebook Status Updates and The Effect of Homophily on the Diffusion of Opinions", CS 144 Ideas Behind the Web.

[8] Erik Boiy; Pieter Hens; Koen Deschacht; Marie-Francine Moens: "Automatic Sentiment Analysis in On-line Text", in Proceedings ELPUB2007 Conference, 2007.

[9] N. Godbole, M. Srinivasaiah, and S. Skiena, "Large-scale sentiment analysis for news and blogs", in ICWSM '07, 2007. Godbole et al.

[10] Alexander Pak, Patrick Paroubek: "Twitter as a Corpus for Sentiment Analysis and Opinion", CS 144 Ideas Behind the Web Team: Dai Wei, Doris Xin Mining.

[11] Galuba, W., Aberer, K., Chakraborty, D., Despotovic, Z. and Kellerer, W. (2010) "Outtweeting the Twitterers: Predicting Information Cascades in Microblogs", Boston, MA.