ISSN:0975-9646

Sentiment Analysis of Product Reviews

Sanket Kulkarni
Department of Information Technology, Pune Institute of Computer Technology
Pune, India

Neha Prabhune
Department of Computer Science, Pune Institute of Computer Technology
Pune, India

Vasanti Sathe
Department of Computer Science, Pune Institute of Computer Technology
Pune, India

***Abstract*— The paper gives a detailed report of sentiment analysis on reviews of a any product using Natural language processing techniques. It gives details about the entire workflow and applications of sentiment analysis. A methodology is proposed to analyze the product reviews to help designers gain insights about the general opinion of their product. Methods of scraping online customer reviews have been explained. Information retrieval and processing of retrieved textual data has been described briefly. Sentence based categorization is considered primarily for extracting the sentiment from the reviews.**

***Keywords*—** *Sentiment Analysis, Feature Extraction, Natural Language Processing, Opinion Identification.*

I. Introduction

In today's world, online trading and online shopping have been one of the major business forms. These websites provide their customers with a review section wherein the customers provide their feedback regarding their purchased products. Increasing number of internet users increases the customer opinions which in turn enlarges the publicly available web data. Sentiment analysis or opinion mining is a way to understand customers opinion about a specific product or various products. Sentiment analysis makes use of this online data and outputs the exact judgment various customers make of the product. The result is beneficial for customers who are about to purchase the product as well as the manufacturers who would design their future products accordingly.

In this paper, we have performed feature based sentiment analysis on a mobile phone. We have gathered different reviews about a particular mobile phone from various websites. This is achieved by using web scraping. The further process involves natural language processing on the collected data. Our process involves removal of stopwords forwarded by parts-of-speech(PoS) tagging. From the tagged structure of the reviews, selective words of particular POS tags are extracted. Product attributes or feature identification and selection is based on these selected words. Certain machine learning algorithms are applied on these reviews to classify them as positive, negative or neutral. The final output is in the form of ranked reviews.

II. Related Works

There are two different approaches to the task of sentiment classification, one approach based on lexicons while other based on machine learning algorithms. Pang et al. [8] have demonstrated classification of movie reviews using machine learning algorithms like Naive Bayes, Maximum Entropy and Support Vector Machines. It was well established by their experimental results that SVM approach outperformed all other methods. In [9], Prabowo and Thelwall considered the movie as well as product reviews for their experiments. They used a combination of supervised learning algorithms and rule-based classification. Machine Learning approach is efficient only when large labeled corpora are available for the tasks of training and validation.

To classify the sentiment of the text, the lexicon-based approach analyses the opinion words in the text and measures the polarity of the text by making use of lexicons like SentiWordNet and WordNet[11]. Turney et al [2] determined the orientation of sentiments based on predefined words. Their approach classifies a document as positive or negative by considering the average semantic orientation of phrases. Morinaga et al [3] experiments on sentiment classification by comparing reviews of different products in a particular category. However, it does not involve summarization of reviews. Their work involves mining frequent phrases like "doesn't work" or "no problem", whereas our work involves mining product features. On the lines of phrase finding, Tong[4] generates timelines of sentiment. They track various discussions of movies available online and displays the positive and negative reviews. Phrases like "great acting" or "uneven editing" are manually added to a list with a positive or negative tag. We manually tag only the adjectives. In [5], Kennedy and Inkpen have implemented the review classification task by calculating the number of positive and negative terms.

Before finalising a particular approach, our team carried out and tried various other methodologies. These were as follows:

1) Bag of Words:

This involves the creation of a matrix having columns as the terms in the dataset whereas the rows are the documents, or the sentences of the review database in our case. This is the most common method used to convert textual problem to a numerical problem which makes it easy for machine learning models. This is bag of words, which is vectorised to and is fitted to the classifier. Usually, the Naive Bayes classifier has proven to show better performance than others with regards to this approach.

2) Word Cloud:

This approach may be used when number of features may not be specified for grouping. We found that reviews cannot be linked properly using this approach. Word Clouds can be used to plot or visualise the important terms. However, as shown in the below figure, there are many outliers also occurring and hence, insights gained are not beneficial.

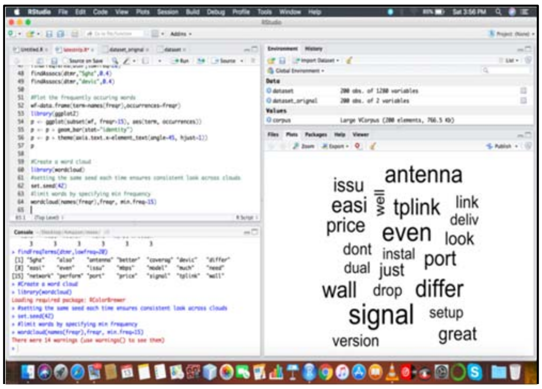

Fig 1. Word Cloud for reviews of 'router'

3) TfIdf:

This method can also be deployed in scenarios where textual content is converted to numerical form which enables the use of machine learning models. Through this method, more insights are particularly gained into the classification criteria. TF stands for term frequency, which indicates the overall importance of any term in a document. IDF, inverse document frequency, measures how important a term is. While computing TF, all terms are considered equally important. However it is known that certain terms, such as "is", "of", and "that", may appear a lot of times but have little importance. Thus we need to weigh down the frequent terms while scale up the rare ones, by computing IDF. Together, tfIdf is a statistic that is intended to reflect how important a word is to a collection of documents in a collection or corpus. The limitation with this approach is that it may give importance to rarely occurring terms and affect the performance of the system. Further, Cosine Similarity was used to plot the figure below. The thickness of the edges shows the strength of the relation between the reviews. More the thickness, more the probability of the reviews belonging to the same category.

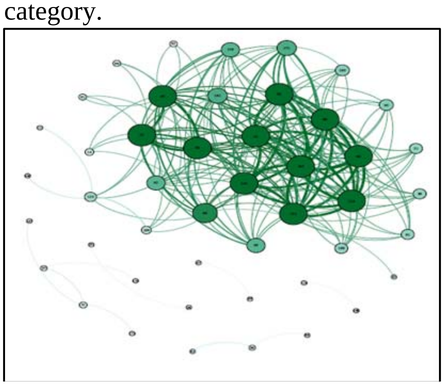

Fig 2. Linking of reviews through cosine similarity

In our paper, we present a classification based on the association of adjectives with the product features. Previous work of Bruce and Wiebe,2000 [6] have successfully established a positive and statistically significant correlation between the accuracy of classification with the presence of adjectives. Liu[1] represents a work which is closest to our work. However, he has defined a feature-based sentiment analysis model(object, feature, opinion, opinion holder, time) which is expressed as an opinionated document. We do not make use of such a feature vector in our implementation.

III. Proposed System

The workflow of the proposed system is given in figure 1. The system takes in a Web page as an input which contains all the reviews of a particular product. The system gives output in the form of the percentage of a feature being positively reviewed, negatively reviewed or neutral. The system contains the following steps for the summarization task: 1) Preprocessing review dataset; 2) Tokenizing the reviews; 3)POS Tagging; 4) Feature identification; 5) Opinion generation.

At first, the system crawls the reviews based on the given input webpage and puts it into a review database. Next step is to find out the frequent features that are been expressed by people in the reviews. Specific opinion words are extracted from these features and their semantic orientation is identified with the use of WordNet and SentiWordNet. Lastly, each opinion sentence is analyzed for its orientation and a brief summary is produced.

Preprocessing of the dataset involves removal of all stopwords are all those common words which are not very helpful while selecting documents or text which match a user query. Consider a review:

"Old stock and SIM 2 not working."

After removal of stopwords, preprocessed review looks like this: "Old stock SIM 2 not working." Our system makes use of stopwords list present in the corpus module of "nltk" package. The preprocessed dataset contains a sequence of all reviews, placed one after another. Next step is to tokenize this text file into individual sentences. PunktTokenizer is a sentence boundary detection algorithm available in "nltk" is used to extract individual sentences. This tokenizer is a pre-trained model for the English language.

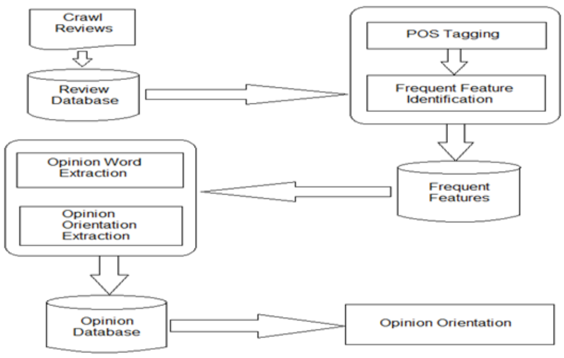

Fig 3. Architecture diagram of the proposed system

In this paper, our focus is on the key features of the product in the reviews. These features are either nouns or noun phrases. Thus, POS tagging is one of the most important tasks of the entire process. Pos_tag function in "nltk" library is used for this purpose. Below is an example of a sentence with POS tags.

[(u'Old','NNP'),(u'stock','NN'),(u'and','CC'),(u'SIM','NNP'),(u'2','CD'),(u'not','RB'),(u'working','VBG')]

Pos_tag generates a key, value pair which is the word, pos tag. In the above example, 'NNP' stands for a noun phrase and 'NN' stands for a noun. Each sentence is saved in a new database which contains POS tagged word in that sentence.

The task of our system is to find out what people like and dislike about the product. Thus, finding product features is a very important step. Consider an example review sentence. "Best phone, great camera and good battery backup." In this sentence, the user seems very happy with the phone quality, camera, and battery. There are various ways in which frequent features are extracted. The system makes use of calculating the frequency of nouns or noun phrases in all sentences. All those words which have a frequency greater than one are considered as frequent features. The function designed contains noun phrases which contain multiple noun words which can be considered as a product feature. An example of the result of this process is shown below.

[(u'PHONE',42),(u'CAMERA',14),(u'battery life',3)]

'PHONE' word is used in 42 reviews, 'CAMERA' is used in 14 reviews whereas 'BATTERY LIFE' is used in 3 reviews.

After extracting the frequent features, the system identifies opinion words. Opinion words are those words which express subjective opinions from sentences. Opinion words are generated based on the frequent features, words contained by the review dataset and a pre-defined set of negative words. The negative wordset includes words like "never", "don't", "ain't" and many more.

Algorithm for opinion word extraction

```
for each feature in frequent features
    for each word in review dataset
        for each subword of a feature
           if subword in word
               subword is an opinion word
```

Now, for each of the opinion words, we identify its orientation. This is achieved with the help of available synsets like WordNet and SentiWordNet. First, the opinion word is sent as an input to Wordnet. A function named synsets obtains all the words in close proximity to our input. Next step is to send the obtained synsets to the function of SentiWordNet which is responsible for allocation of positive and negative scores. Orientation of the opinion word is decided in the following way:

```
if positive score>negative score
   return positive
if negative score>positive score
   return negative
```

Previous work clearly indicates a positive correlation between subjective opinion and presence of adjective. Thus, while calculating the orientation of the word, a check is done on its POS tag. Our work keeps track of all the nearby adjectives and adverbs by checking their POS tags. A count is maintained for all such words which are associated to a single product feature. Lastly, each product feature is assigned a positive, negative and neutral value which is calculated on the basis of the orientation of the words and count of opinion words. Final output format is represented as shown below:

[u'CHARGER':[31.0,0.0,69.0]]
[u'CAMERA’:[25.0,7.0,66.0]]

It tells us that there are 31% positive reviews, 0% negative reviews and 66% neutral reviews about the charger whereas

there are 25% positive reviews, 7% negative reviews and 66% neutral reviews about the camera.

IV. Performance Analysis

An opinion mining system has been implemented based on the methodologies discussed above. We shall evaluate the system from the perspective of its accuracy in determining accurate opinion identification for the features of a product. We have fed input to the system in the form of user reviews for two products, 1 for cellphone and the other for a router. These reviews were collected from amazon.in.

For creating our review database, we scraped all the user reviews available for the particular products by writing a crawler in python. Other information available such as rating were not required in this context. For the purpose of evaluation, we manually went through and evaluated all the reviews. Each sentence of the review was considered when the opinion about a feature of the product could be identified. In this manner, a manual dictionary for the feature set was created. We manually cross checked the results obtained through human intelligence and that obtained by our system.

A small obscurity arises when the features occur as implicit features. For eg: in the sentence, “it can be carried around anywhere easily”, the feature being discussed is the mobility or the size of the product. Such features which do not appear in the reviews are implicit features. However, most of the reviews contain explicit features hence this complication can be considered as a minor one, with respect to the performance.

After interpreting the results for performance evaluation, we found out that our system provides a good accuracy of 84.61% in determining the overall orientation of the opinions. Hence in summary, we can conclude that our system provides a promising approach for opinion identification based on features for a particular product. We also wish to share some limitations of the proposed system: 1) As stated, implicit feature identification has caused a minor hurdle. 2) We have primarily made use of adjectives and adverbs for opinion identification. This can however be extended to making use of verbs. For eg: in the sentence “I highly recommend the phone”, verbs can be made use of.

V. CONCLUSION AND FUTURE WORK

In this paper, we have proposed our technique to summarize product reviews based on natural language processing mechanisms. The main objective was to provide a feature based summarization of the available customer reviews. Performance analysis clearly indicates that our techniques show very promising results. This is a highly important task as summarizing reviews is very important for the product manufacturers.

In our future work, we plan to refine our methods for even better results. We will find ways to determine the strength of each opinion. We will also work on generating a textual overview of all the reviews which would cover maximum sentiments expressed in the reviews. We do believe that this particular task would be beneficial not only for the manufacturers but also the product designers.

REFERENCES

[1] Hu, M., and Liu, B. 2004. Mining Opinion Features in Customer Reviews. *AAAI’04,* 2004.

[2] Turney, P. 2002. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. *ACL’02*.

[3] Morinaga, S., Ya Yamanishi, K., Tateishi, K, and Fukushima, T. 2002. Mining Product Reputations on the Web. *KDD’02*.

[4] Tong, R., 2001. An Operational System for Detecting and Tracking Opinions in on-line discussion. *SIGIR 2001 Workshop on Operational Text Classification.*

[5] Kennedy A, Inkpen D (2006) Sentiment classification of movie reviews using contextual valence shifters. *Computational Intelligence* 22(2):110-125.

[6] Bruce, R., and Wiebe, J. 2000. Recognizing Subjectivity: A Case Study of Manual Tagging. *Natural Language Engineering*.

[7] Cardie, C., Wiebe, J., Wilson, T. and Litman, D. 2003. Combining Low-Level and Summary Representations of Opinions for Multi-Perspective Question Answering. *2003 AAAI Spring Symposium on New Directions in Question Answering.*

[8] Pang, B., Lee, L., and Vaithyanathan, S., 2002. Thumbs up? Sentiment Classification Using Machine Learning Techniques. In *Proc. of EMNLP 2002*.

[9] Prabowo R, Thelwall M (2009) Sentiment analysis: A combined approach. *Journal of Informetrics*, 3(2): 143–157.

[10] Tang H, Tan S, Cheng X (2009). A survey on sentiment detection of reviews. *Expert Systems with Application* 36(7):10760–10773.

[11] Miller, G., Beckwith, R, Fellbaum, C., Gross, D., and Miller, K. 1990. Introduction to WordNet: An on-line lexical database. *International Journal of Lexicography (special issue)*, 3(4):235-312