Information Extraction & Opinion Analysis Using NLP

Shubham Mehrotra
Indian Institute of Information Technology
Allahabad
India
+91-9936013475

Ronish Kalia
Indian Institute of Information Technology
Allahabad
India
+91-9005800306

Pankaj Wadhwani
Indian Institute of Information Technology
Allahabad
India
+91-

Shubham Sharma
Indian Institute of Information Technology
Allahabad
India
+91-9559039231

Dhruv Kumar
Indian Institute of Information Technology
Allahabad
India
+91-9559039231

Arjun Banga
Indian Institute of Information Technology
Allahabad
India
+91-8795089959

Dr. Ratna Sanyal
Indian Institute of Information Technology
Allahabad
India
+91

1. Abstract

An important part of our information-gathering behavior has always been to find out what other people think. With the growing availability and popularity of opinion-rich resources such as online review sites and personal blogs, new opportunities and challenges arise as people now can, and do, actively use information technologies to seek out and understand the opinions of others. ” With this paper, we aim to feel the pulse of the E commerce market. What we generally find in the reviews section of these websites is rather deceptive. The star rating of the user is not always concordant with what he/she has written. Also there is no feature wise review available for products. We aim to bridge this gap with our project, make the customers absolutely sure with what they are buying and hence make the E commerce market more user friendly.

2. *Introduction*

The goal of the project, is to develop a prototype that can feel the pulse of the E-Commerce website users with regard to the reviews they provide on the products they buy. This is done using the Opinion Analysis Techniques (a form of NLP).
Opinion analysis aims to determine the attitude of a speaker or a writer with respect to some topic or the overall contextual polarity of a document. The attitude may be his or her judgment or evaluation, affective state (that is to say, the emotional state of the author when writing), or the intended emotional communication. It is the computational study of opinions, opinions and emotions expressed in text.

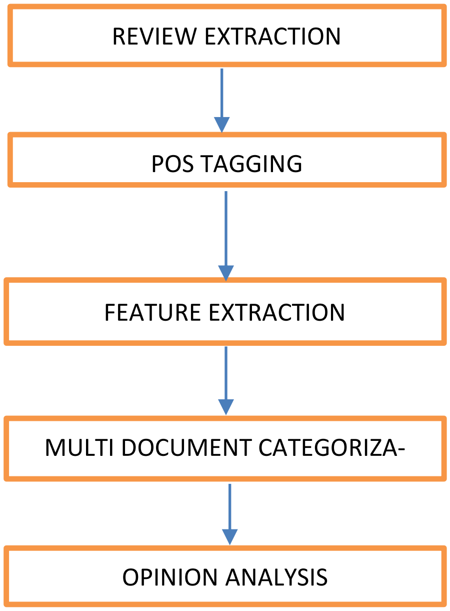
3. Objective

Our first task is the extraction of the information i.e. the reviews on the products from the e-commerce website and then their analysis.
We will be using a Scripting Language to convert a series of HTML table files to CSV and Parser Tools to deploy natural language processing techniques to automate opinion analysis on large collections of texts, including web pages and online reviews which can be further deployed on web blogs, social media, online news, internet discussion groups etc.

After we have got the reviews and the features extracted, we have used python libraries such as nltk, numpy, yaml etc. to give scores to the seed words (taken initially) and their corresponding synonyms present in the dictionary after which we have calculated the final score for the document.

Then we have created the Graphical User Interface (GUI) in Java where the user can enter the product id and the output will show him the final computed rating of the product. He would also be able to see the rating of some of the top features of that product.

4. Our Proposed Theory



4.1. Review Extraction

1. Amazon comment pages follow a particular type of URL: **http://www.amazon.com/product-reviews/"ProductID"/?ie=UTF8&show Viewpoints=0&pageNumber="PageNo"&sortBy=bySubmissionDateDescending**

2. So, given a Product Id of any amazon product, its reviews can be found at the above url, and changing the "PageNo" will lead us to different pages having different comments. Note that page no can only be a natural number.

3. Now, since we have the url of a webpage, its trivial to extract the HTML source code of the webpage. Now, the problem is to extract comments from a webpage, given its HTML source.

4. This problem can easily be solved using pattern matching. Reviews from any HTML source code can be found after the given block using the following pattern matching in Java:
<divclass = "reviewText"> ([^<]*?) <\/div> .*? <div style = "padding-top#gs

5. Moreover, Star Rating can be found after the following block using the following pattern matching in JAVA:
<div.*?star_([1-5])_([05]).*?\<b\>(.*?)\<\/b\>.*?br\>

6. Star rating and comments is stored in csv file. Reviews can be stored iteratively until no more comments are there on the page.

7. Next page can then be analyzed for more comments in a similar manner.

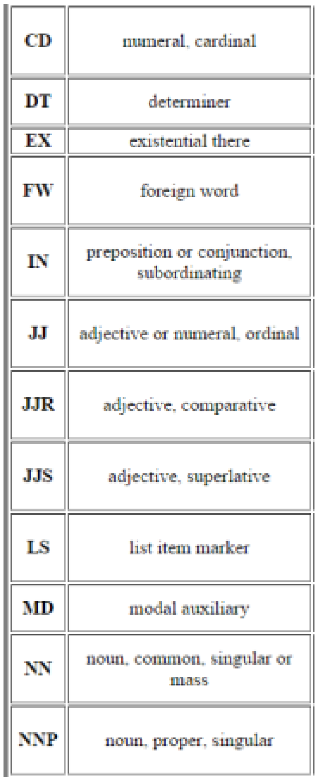
4.2. POS Tagging

The features were decided based on the frequency of occurrences in the data. For each particular feature we will find synonyms of those features from the WordNet and store them.

Then in each sentence we will look for those particular features or their synonyms and then categorize those sentences under those features. Now for each categorized sentence we would apply opinion analysis.

After we have applied the opinion analysis we would then rate every feature according to the results obtained from the opinion analysis.

(Ref:- http://nlp.stanford.edu/software/tagger.shtml)



CD	numeral, cardinal
DT	determiner
EX	existential there
FW	foreign word
IN	preposition or conjunction, subordinating
JJ	adjective or numeral, ordinal
JJR	adjective, comparative
JJS	adjective, superlative
LS	list item marker
MD	modal auxiliary
NN	noun, common, singular or mass
NNP	noun, proper, singular

4.3. Feature Extraction

This model first discovers the targets on which opinions have been expressed in a sentence, and then determines whether the opinions are positive, negative or neutral. The targets are objects, and their components, attributes and features. An object can be a product, service, individual, organization, event, topic, etc. For instance, in a product review sentence, it identifies product features that

have been commented on by the reviewer and determines whether the comments are positive or negative. For example, in the sentence, "*The battery life of this camera is too short*" the comment is on "battery life" of the camera object and the opinion is negative. Many real-life applications require this level of detailed analysis because in order to make product improvements one needs to know what components and/or features of the product are liked and disliked by consumers.
An *object o* is an entity which can be a product, person, event, organization, or topic. It is associated with a hierarchy of *components* (or *parts*), *sub-components* and *A* is a set of *attributes* of *o*. Each component has its own set of sub-components and attributes.

Ex: A particular brand of cellular phone is an object. It has a set of components, e.g., *battery*, and *screen*, and also a set of attributes, e.g., *voice quality*, *size*, and *weight*. The battery component also has its set of attributes, e.g., *battery life*, and *battery size*.

Using features for an object is quite common in the product domain as people often use the term *product features*. However, when the objects are events and topics, the term *feature* may not sound natural. We choose to use the term *feature* along with the term *object*.

Finding frequent nouns and noun phrases: Nouns and noun phrases (or groups) are identified by using a POS tagger. Their occurrence frequencies are counted, and only the frequent ones are kept. A frequency threshold can be decided experimentally. The reason for using this approach is that when people comment on product features, the vocabulary that they use usually converges, and most product features are nouns. Thus, those nouns that are frequently talked about are usually genuine and important features. We will also consider only the cases where the Nouns or Nouns Phrases are followed by Adjectives or Adverbs because the adjectives/adverbs are generally used to enhance the list of features. Irrelevant contents in reviews are often diverse and thus infrequent, i.e., they are quite different in different reviews. Thus, those nouns that are infrequent are likely to be non-features or less important features.

We have scanned the reviews with the help of POS tagger. We have tagged nouns, noun phrases, adjectives and the other parts of speech from the reviews. To select our list of features, we have maintained a priority queue containing nouns/noun phrases according to their respective frequencies of occurrence. Then we will extract the most frequently occurred nouns/noun phrases from the priority queue.

Features	Frequency
Camera	87
Quality	32
Time	17
LCD	14
Video	12
Flash	10
Battery	9

Fig 2: A list of features extracted
for a Canon Digital Camera.

4.4. Multi Documented Categorization of Reviews

Before this section begins all the features have already been extracted by us. For each particular feature we will find synonyms of those features from the WordNet and store them. Then in each sentence we will look for those particular features or their synonyms and then categorize those sentences under those features. Now for each categorized sentence we would apply opinion analysis. After we have applied the opinion analysis we would then rate every feature according to the results obtained from the opinion analysis.

4.5. Algorithm used for Opinion Analysis

To generate a dictionary of words we will use a small initial seed lexicon of positive, negative and neutral words. These words are expanded through synonyms and antonyms link in word net.

Part of speech is applied to the initial dictionary to distinguish b/w multiple word sense. Score of each word present on dictionary is positive if the word is positive, negative if the word is negative and zero if the word is neutral.

Vector s^m = score of each word present in the dictionary.
$S_i^0 = 1$ if w_i is positive, -1 if w_i is negative and 0 for neutral.

After that a matrix A is generated, whose jth element of the ith row is 1, if i equals j, $+\lambda$ if jth word is a synonym of the ith word, $-\lambda$ is jth word is an antonym of ith word and 0 otherwise.
We then propagate the opinion scores by repeated multiplication of the calculated matrix A against score matrix.

```
For m : 1 to M
        $S^m = \text{signCorrect}(As^{m-1})$
```

We will try different values of λ, and stick to the one with highest accuracy.

For opinion analysis, we will calculate raw score of each sentence.

Consider x to be a sentence consisting of n words $w_1 + w_2 + \dots + w_n$ -
RawScore(x) = $\sum s_i$ for i : 1 to n

Now, suppose that there are two sentences having 2 words each.

First sentence having words with score +3 and -1, producing Rawscore = 2.
Second sentence having words with score +1 and +1, producing Rawscore = 2.

But since the second sentence is better as it is having more number of positive words.

A metric has to be defined to differentiate b/w equal Rawscore sentences, that metric is Purity.

$$Purity\ (x) = \frac{Rawscore(x)}{\sum_{i=1}^{n} |\mathrm{si}\,|}$$

Based on the purity and rawScore of every sentence in a review, the review can be judged.

After we have the score for all words in the document, we calculate the final score using the following approach.

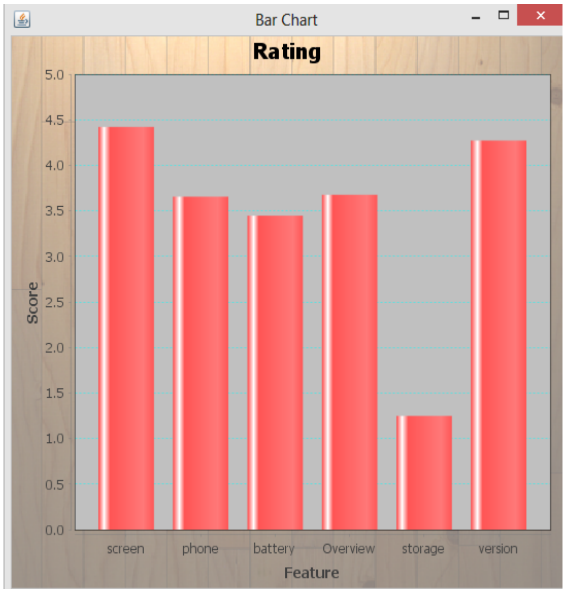
Word	Score
Hurt	-8.44168
Anguish	-5.46496
Blur	-5.09159
Boost	5.46496
Learn	6.33901
Advance	6.95328

Evaluation Approach:-

The overall opinion is measured by the rating given to it on a scale of 1 to 5.
It is calculated as:
Rating = ((NUM/DEN) + 1.0)* (5.0/2.0)
Where:
NUM denotes the sum of score of each word.
DEN denotes the sum of absolute value of the score of each word.
The score of each word could be a positive or negative value depending on the opinion it is used for. Words carrying negative opinion

are given a negative score whereas words carrying a positive opinion are given a positive score.

5. Results



Final Results

6. References

[1] http://en.wikipedia.org/wiki/Information-Extraction

[2] Alekh Agarwal and Pushpak Bhattacharyya, Opinion Analysis: *A New Approach for Effective Use of Linguistic Knowledge and Exploiting Similarities in a Set of Documents to be Classified*, International Conference on Natural Language Processing (ICON 05), *IIT Kanpur, India, December, 2005.*

[3] Alistair Kennedy and Diana Inkpen, Opinion classification of movie and product reviews using contextual valence shifters, *Computational Intelligence, 22(2):110–125, 2006*.

[4] Balamurali A.R., Aditya Joshi, Pushpak Bhattacharyya, Robust Sense Based Opinion Classification, *ACL WASSA 2011, Portland, USA, 2011*

[5] Daniel M. Bikel, Jeffrey Sorensen, If We Want Your Opinion, *International Conference on Semantic Computing (ICSC 2007), 2007.*

[6] Farah Benamara, Carmine Cesarano, Antonio Picariello, VS Subrahmanian et al; Opinion Analysis: Adjectives and Adverbs are better than Adjectives Alone, *In ICWSM '2007 Boulder, CO USA, 2007.*

[7] Luhn, H. P., The automatic creation of literature abstracts, *IBM Journal of Research Development. 2(2):159-165, 1958.*

[8] Marcu, Daniel, The Theory and Practice of Discourse Parsing and Summarization, *MIT Press, Cambridge, MA, 2000.*

[9] Ng, Vincent and Dasgupta, Sajib and Arifin, S. M. Niaz, Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews