Available online at http://www.journalcra.com

International Journal of Current Research
Vol. 9, Issue, 11, pp.60419-60422, November, 2017

INTERNATIONAL JOURNAL OF CURRENT RESEARCH

ISSN: 0975-833X

RESEARCH ARTICLE

INFORANK AN APPROACH FOR REVIEW RANKING

***,[1]Akshit Bhatia, [2]Kanika Mittal, [3]Natansh Negi and [4]Vaibhav Tomar**

[1](B. Tech, (CSE)) Bhagwan Parsuram Institute of Technology, Delhi, India
[2]Assistant Professor (CSE-Dept.)-BPIT, Delhi, India
[3](B. Tech, (IT)), Maharajah Surajmal Institute of Technology, (B. Tech, (CSE))
[4]ABES Engineering College, Ghaziabad, India

ARTICLE INFO

Article History:
Received 20th August, 2017
Received in revised form
03rd September, 2017
Accepted 10th October, 2017
Published online 30th November, 2017

Key words:

Ranking of the Reviews, Information content of review, Ranking based on the information available.

ABSTRACT

Reading numerous reviews about a product often is a cumbersome work. Many times, the review only say that the product is good and nothing else about the design, functionality of that product and hence reading those types of reviews doesn't influence the decision of the customers. So to make the life of a customer easier, this paper presents an algorithm that ranks the reviews on the basis of the amount of information in the review. The earlier works done on the ranking of reviews assumed that all features were equally important and hence the ranking was never based on the amount of information that can be extracted from the review. This algorithm is an improvement over the other works in the similar domain as it extracts the features based on their importance among the users and then ranks the review as per the information available in the review.

Citation: Akshit Bhatia, Kanika Mittal, Natansh Negi and Vaibhav Tomar, 2017. "Inforank an approach for review ranking", *International Journal of Current Research*, 9, (11), 60419-60422.

INTRODUCTION

Nowdays, the impact that a review of a product can bring on a customer can't be ignored. Nearly, every product is today sold based on the reviews about it by the peer customers. Michael Aldrich introduced online shopping which did prove to be a success. The budget of the customer, customer's reviews, the after sales service and the return policy plays an important role in attracting customers. This research work concentrates on the contents of a review rather than the votes that a review gathers through other users. If a user wants to purchase a mobile phone, he mightgoogle the query "ABC mobile". But his decision to buy a mobile will be based to some extent, on the experience other customers have. This is what also referred to as 'word of mouth'. However, an enormous number of reviews with many, having little or no information about the product would make the customer's experience bad. So for the online sellers like Amazon, Flipkart or eBay who don't want their customers to go through heaps of useless comments, this algorithm would help them go through only the informative reviews and hence figure out their decision. This algorithm outputs the reviews based on the number of the features that the review describes. The equations (1) and (2) that are used in *Section III* resemble Shannon Theorem's formula that is used to derive the entropy of a system. This entropy is the total score of the review. The first step in review ranking is to create a dictionary of all the features that influence the decision of the customer. Here feature can refer to the design of a product, functionality of that product or build quality. Unique weights are assigned to each feature. The maximum weight, assigned to a feature, is equal to the total number of features considered in the dictionary. Then the review is read and if there is a match between a feature from the review and the feature from the dictionary created above, then that feature's weight is used for the calculation later. The second step is to measure the orientation (semantic) of each phrase or sentence (Hatzivassi-loglou and McKeown, 1997). A phrase is considered to be good when it has good resemblance (e.g., "amazing") and bad when it has a bad resemblance (e.g., "worst").More specifically, the mutual information between the given phrase and the word "good" minus the mutual information between the given phrase and the word "bad" gives us a numerical rating for ranking. The algorithm is presented in the later section.

Corresponding author:* **Akshit Bhatia,
(B. Tech, (CSE)) Bhagwan Parsuram Institute of Technology, Delhi, India.

Related Work

Reviews Ranking

Hatzivassiloglou, Wiebe and Wiebeetal considered the refinement of reviews by assigning semantic orientation to each objective i.e. a positive adjective or a negative adjective (Hatzivassiloglou and Wiebe, 2000). Revrank Algorithm by Oren Tsur and Ari Rappoportconsidered creation of a virtual core which contains the features according to and for each feature if present in the review, its score relative to the length is calculated (Oren Tsur *et al*., 2009). Peter D. Turneyclassifies each review as if it recommends the product or if it criticises the product by calculating semantic orientation of the review (Peter D. Turney, 2002). Richong Zhang and Thomas Tran used Shannon's approach to extract the amount of information which may be available to the user (Richong Zhang and Thomas Tran, 2008).

Product Ranking

Kunpeng Zhang, Ramanathan Narayanan and Alok Choudharyrank products based on online reviews by segregating the review sentences in two broad categories and by making a graph and giving the edge weights and node weights (Kunpeng Zhang Ramanathan Narayanan and AlokChoudhary, 2010). The technique used in this paper, uses a different method for edge and node weight assignment. There are also works that help in automatic extraction of product features from online reviews (Ana-Maria Popescu and Oren Etzioni, 2005). "Mining Millions of Reviews" ranks products based on how useful a review is (Kunpeng Zhang *et al*., 2011). The research paper Y. Lu, C. Zhai, and N. Sundaresan by on summarising short comments uses techniques of topic modelling under which broad category, our research paper also lies (Lu *et al*., 2009).

Ranking algorithm

Overview

The algorithm proceeds to find out the information contained in each review. It identifies the most important part of a review. The first task is to create a list of features that a customer expect, a product should have. Then, assign certain weights to them according to their demand. The second step is to extract features described in the review If a feature has been included it would not be included again. This step ensures that a review is not ranked only on the basis of one feature. The next step of the algorithm is to apply a positive or negative semantics to each selected feature. Applying the following formula would give us a term's score in the review. It may be considered as its contribution to the review. The equation (1) is for reviews larger than a threshold. If the review is larger than threshold value then the equation (2) is used. The threshold is decided based on the number of features of a product. If the total number of features possible in the review are 'x', then the long review is considered if the review has '2x' number of lines. Similarly a short review may correspond to less number of features matched from the review (we considered x/2 features here)

$$P(i) = -w(i)/L * log(w(i)/L) \quad (1)$$

$$P(\mathrm{i}) = -w(i)/L * \log(w(i)/10L) \quad (2)$$

The above formulas firstly calculates the score of each feature, 'P(i)' according to its weight and then if the review is too long, the variable L (length of the review based on the number of times all the features are described in the review) would decrease the score of the review. For a small review division by 10 would again decrease the score of the review. The variable 'i' here refers to the index of a feature being considered. To calculate the entropy of a review, equation (3) is used.

$$H(\text{review}) = -\sum P(i) * logP(i) \quad (3)$$

This is the total score of the concerned review. Lets call this score as '*entropy (review(i))*'.As the logarithm of a decimal value yields a negative value, the negative sign in equation (1) and (2) will neutralise the sign. Similar is the inversion when the log of the value (P(i)) is calculated. The next step is to calculate the amount of information. First the total score that is possible for the review, when all the features of a product present in the dictionary are considered, is calculated. The individual score of a feature is calculated through equation (1) or equation (2). The review's score is then calculated with the help of equation (3), lets call this total score as '*full_entropy(review(i))*'. This new score is then divided by the total score possible in the review and in this way the percentage of information present in the review is calculated. Equation (4) is used here. To find out if the review is describing the product as good or bad, the semantic/polarity of each sentence is found out. Here only those sentences are considered which contains a feature matched from the dictionary. The positive sentence is assigned a positive weight(where weight refers to w(i) in equation (1)) while the negative sentence is assigned a negative weight. After adding the positive scores with a negative scores, one would get value which would tell if a review is negative (if the value has minus sign) or positive(if the value has plus sign).

$$info = \frac{\text{entropy}(review[i])}{\text{full\_entropy}(review[i])} * 100\% \quad (4)$$

Algorithm

1. Determine the features of a product(like battery, or RAM or camera etc. of a mobile phone) that customers usually describe in their reviews and assign unique weights to each feature according to their importance (weights here are integers and are assigned in the range of 0 to total number of features.)
2. Calculate each feature's score by using equation(1) or equation (2).
3. Semantics of Review (positive or negative): The method: demo_liu_hu_lexicon under the package: nltk.sentiment.util is used to calculate the semantic of a sentence.
4. Rank calculation: percentage of information involved in the review is calculated using equation (4). This describes to what measure does a reviewer describes the product. And arranging the scores of different reviews in a decreasing order, one would get the ranked list of reviews from the most informative review on top and a stupid review at bottom.

Unlike other algorithms, this algorithm is intended to solve two problems, one to calculate scores of reviews with different feature weighted as per their relevance among customers and,

second both measuring the product quality and ranking of a particular review is possible. If the score of a review is 0 then it can be concluded that the review is a spam.

consumers finish their information search and decision making process easier. In comparison to other model, this algorithm is easier to understand and hence easier to implement. The

I.Category assignment

Percentage of information	Category
50% to 100%	Informative
35% to 50%	Average
0% to35%	Poor

II. Compiled result on Samsung neo

Reviews	Semantic orientation calculated manually	Semantic orientation using Algorithm	Percentage of information	Category	Judge's category
1	Positive	Positive	48	Average	Average
2.	Positive	Positive	56.5	Informative	Informative
3.	Positive	Positive	32.5	Poor	Poor
4.	Positive	Negative	24.4	Poor	Average
5.	Positive	Positive	43.3	Average	Average
6.	Positive	Positive	28	Poor	Poor
7.	Positive	Positive	50	informative	average
8.	Neutral	Neutral	0	Poor	Poor
9.	Positive	Positive	25	Poor	Poor
10	Positive	Positive	10	Poor	Poor

III. Compiled result on reviews of Sony experia

Reviews	Semantic after calculation	Semantic orientation calculated manually	Percentage obtained	Category	Judge's type
1	Positive	Positive	17	Poor	Average
2.	Positive	Positive	41.2	Average	Average
3.	Positive	Positive	41.3	Average	Average
4.	Positive	Positive	82	Informative	Informative
5.	Negative	Negative	24	Poor	Poor
6.	Positive	Positive	41.5	Average	Average
7.	Negative	Negative	30	Poor	Average
8.	Positive	Positive	80	Informative	Informative
9.	Positive	Positive	81.5	Informative	informative

RESULTS

Reviews are collected from online shopping giants like flipkart.com, amazon.com, snapdeal.com etc and reviews from several 3rd party blogsites like buyingiq.com. After going through the several reviews, we extracted features that are most talked among the users and they are weighted based on their importance. In a dictionary, we keep those important words with their synonyms and the score that we gave to each. Out of 50 reviews that we selected, we found the semantic orientation of 45 reviews to be predicted correctly. Similarly, out of these reviews, we could determine the category of 47 reviews to be correct. We compared our result on a review with the help of one judge who marked the reviews as Average, informative or neutral independently of the results published in the table. Also out of 50 reviews, we were able to predict the ranking of 45 reviews. Here again, we took the help of a person unaware of the results that the algorithm has produced. Here in Table I, category of a review is divided into 3 parts. In Table II and Table III, an example of the model is presented which compares the model's prediction with the judge's evaluation of a review.

Conclusion and Future Scope

This paper proposes a way to enable customers only go through the informative reviews rather than useless reviews. The implementation would enable sellers to rank review based on the characteristic of the product described rather than the number of votes that a review receives. This will help algorithm can classify and rank reviews easily and quickly. 45 out of 50 reviews being ranked correctly. This concludes that this algorithm works great. Reviews form other product categories and much larger review sets will be investigated in the future. Future research may adopt a improvised feature extraction algorithms and would assign variable weights to them as per their importance. Applying this algorithm to wide section of products would be an innovative way of extending this algorithm.

REFERENCES

Ana-Maria Popescu and Oren Etzioni, Extracting Product Features and Opinions from Reviews, 2005

Hatzivassi-loglou and McKeown, 1997. Effects of adjective orientation and gradability on sentence subjectivity, in procedure of COLING.

Hatzivassiloglou and Wiebe (2000), Wiebe (2000), Wiebeetal (2001) Learning subjective adjectives from corpora, 2001

Kunpeng Zhang Ramanathan Narayanan, AlokChoudhary, 2010. Voice of the Customers: Mining Online Customer Reviews for Product Feature-based ranking.

Kunpeng Zhang, Yu Cheng,Wei-keng Liao, AlokChoudhary, 2011. Mining Millions of Reviews: A Technique to Rank Products Based on Importance of Reviews.

Lu, Y., C. Zhai, and N. Sundaresan, 2009. Rated Aspect Summarization of Short comments.

Oren Tsur and Ari Rappoport, RevRank, 2009. A fully Unsupervised Algorithm for selecting the most helpful book reviews.

Peter D. Turney, 2002. Thumbs up or Thumbs Down? Semantic Orientation Applied.

Richong Zhang and Thomas Tran, 2008. An entropy based model for discovering usefulness in review.

******