Product Reputation Model: An Opinion Mining Based Approach

Ahmad Abdel-Hafez, Yue Xu, Dian Tjondronegoro
School of Electrical Engineering and Computer Science
Queensland University of Technology
Brisbane, Australia
ahmad.abdelhafez@student.qut.edu.au, {yue.xu,dian}@ qut.edu.au

Abstract. Product rating systems are very popular on the web, and users are increasingly depending on the overall product ratings provided by websites to make purchase decisions or to compare various products. Currently most of these systems directly depend on users' ratings and aggregate the ratings using simple aggregating methods such as mean or median [1]. In fact, many websites also allow users to express their opinions in the form of textual product reviews. In this paper, we propose a new product reputation model that uses opinion mining techniques in order to extract sentiments about product's features, and then provide a method to generate a more realistic reputation value for every feature of the product and the product itself. We considered the strength of the opinion rather than its orientation only. We do not treat all product features equally when we calculate the overall product reputation, as some features are more important to customers than others, and consequently have more impact on customers buying decisions. Our method provides helpful details about the product features for customers rather than only representing reputation as a number only.

Keywords: reputation model, opinion mining, features impact, opinion strength

1 Introduction

Many websites nowadays provide a rating system for products, which is used by customers to rate available products according to their own experience. Reputation systems provide methods for collecting and aggregating users' ratings to calculate the overall reputation for products, users, or services [2]. This final rate is very important, as it represents the electronic 'word of mouth' that customers build their trust in a product on. On the other hand, most websites allow customers to add textual reviews to explain more about their opinion to the product. These reviews are available for customers to read, to the best of our knowledge, they are not analyzed and counted in the product overall reputation. Many reputation models have been proposed, but most of them concentrated on user's reputation in C2C (Consumer to Consumer) websites such as eBay.com, while service and product reputation has received less attention. Besides, most of the literature about product reputation models neglected users' reviews and counted users' ratings only. Therefore, their reputation systems did not provide any summaries and details about the weakness and strength points in the product.

In this work we will provide a reputation model for products using sentiment analysis methods. The proposed model generates reputation for a specific product depending on the textual reviews provided by users rather than depending on their ratings because users' ratings do not reveal an actual reflection for the products' features, and they do not provide details for customers about features reputation and about "why" the reputation is high or low. For example, a strict user might give three stars for the product although he does not have a clear negative opinion about the product. On the other hand a more generous customer might have a couple of negative opinions about the product but still give four stars. Additionally, textual reviews can be used to provide summaries about product features reputation in addition to the aggregated value for the product reputation, which can make the reputation system more meaningful rather than being just a number. We calculate features impact by counting how many times every feature is mentioned explicitly in the text reviews, assuming that features that are mentioned more by users are more important for them.

In the rest of this paper, we will demonstrate couple of existing product reputation model in the section II, and in the following sections we will explain equations we use to calculate the reputation value for a product. We will also provide diagrams to show the difference between the results of our reputation calculation method and the regular average method used by most websites to represent the overall product reputation.

2 Related Work

2.1 Reputation Models

Reputation models have been studied intensively by many researchers in the last decade, many of these researches concentrated on user's reputation and some of them have discussed product reputations. One of the most basic works on ratings aggregation analyzed robustness of different aggregators, in particular the mean, weighted mean, median and mode, and proposed that using median or mode is more efficient than using mean [1]. Cho et al. [3] proposed a more sophisticated model, they calculated user reputation and used it in order to calculate weights for different ratings. Moreover, they assumed that some users tend to give higher ratings than others, hence, they calculated rating tendency for users and deducted it from user rating. They used the user's accurate prediction and the degree of his activity to define his level of expertise, and then they used this value to represent user's reputation. This method might not be an accurate way to give different weights for ratings, because a user's reputation should not reduce the weight of his opinion about a product. On the other hand, another promising work introduced by Leberknight et al. [4], discussed the volatility of online ratings, where authors aimed to reflect the current trend of users' ratings, they used weighted average where old ratings have less weight than current ones. They introduced a metric called Average Rating Volatility (ARV) that captured the extent of fluctuation present in the ratings, and then they used it to calculate discounting factor, which is used in weighting older ratings.

2.2 Opinion Mining

Many literatures have focused on extracting useful information from the huge amount of available users' opinions in the internet. Opinion mining was used in many different domains. Business Intelligence is the most popular one, where many studies concentrated on mining customers' reviews for better market understanding [5]. Researchers focused on the sentiment analysis part and represented product reputation as a simple count of positive and negative sentiments [6] [7]. Turney [8], Pang et al. [9], and Kamps et al. [10] provided different methods to determine the orientation of a word as positive or negative. In contrast, Hu & Liu [6] proposed a set of techniques for mining and summarizing product reviews to provide a feature based summary of customer reviews, they searched for frequent noun and noun phrases as candidate features. While Popescu et al. [11] identified parts and features of a product depending on finding relation between noun words and the product class using PMI algorithm [8]. Morinaga et al. [12] were one of the first researchers to introduce a general framework for collecting and analyzing users' reviews in order to find the overall product reputation. They used two dimensional positioning Maps, which contained the extracted opinion phrases and associate products with them. The distance between opinion-phrases and products represents closeness. Their proposed method does not mine product features [6], which might be crucial element in the product reputation analysis. In contrast, Hashimoto & Shirota [13] depended on buzz marketing sites to provide a framework for reputation analysis considering product's features. They attempted to discover the topic of each review as initial step, and then they determined important topics depending on the contribution rate of each topic and the polarity of the messages. Finally, the results are visualized for users. However, the effectiveness of their framework has not been evaluated, and the visualization method used to represent the results has not been perfected. Moreover, they neglected topics with lower contributions which might affect the overall product reputation.

To the best of our knowledge none of the previous work has proposed a convenient method to calculate product reputation, depending on the outcome of mining users' reviews. Most of the available methods represent the reputation as a simple count or average of positive and negative opinions in the reviews. While the convenient represented models depended on users' ratings rather than users' textual reviews.

3 The Proposed Approach

3.1 Definition

A product can be described by a set of features representing its characteristics. Some of the features may be more specific or more general than others. For example, for a specific mobile phone product, the "Mobile Camera" is considered as a general feature, while Resolution, Optical Zoom, Flash Light, Video Recording are more specific features of Mobile Camera. In this paper, we define product features as a hierarchy.

Definition 1 (Feature hierarchy): A feature hierarchy consists of a set of features and their relationships, denoted as $FH = \{F, L\}$, F is a set of features where $F = \{f_1, f_2, ..., f_n\}$ and L is a set of relations. In the feature hierarchy, the relationship between a pair of features is the sub-feature relationship. For $f_i, f_j \in F$, if f_j is a sub-feature of f_i, then $(f_i, f_j) \in L$, which means, f_j is more specific than f_i. The root of the hierarchy represents the product itself, and the first level children are the generic features. In this paper, we assume that the feature hierarchy is available.

Definition 2 (User's Review): R is a set of reviews where $R = \{r_1, r_2, ..., r_m\}$. Every review consists of a number of opinions about different features, denoted as $\forall r_i \in R \;\; r_i = \{(f_{i1}, o_{i1}, s_{i1}), ..., (f_{in}, o_{in}, s_{in})\}$. o_{ij} is the orientation of the opinion; $o_{ij} \in \{Pos, Neg, Neu\}$, which represents positive, negative, and neutral respectively. s_i is the strength of the opinion, $s_i \in \{1,2,3\}$, where 1 represents "weak opinion", 2 for "moderate", and 3 for "strong opinion".

In this paper, we assume that the product features and the opinion orientation and strength to the features in each product review have been determined by using existing opinion mining techniques. The proposed reputation model will generate product reputation based on the opinion orientation information, i.e., this information is available, and is the input to the reputation model. There are different methods that can be used to extract this information [14] [15].

For a specific feature f_j, the set of negative reviews are denoted as $R_j^{neg} = \{r_i | o_{ij} = Neg\}$, $R_j^{neg} = \left\{r_1^{neg}, r_2^{neg}, ..., r_{|R_j^{neg}|}^{neg}\right\}$, where $\forall r_i^{neg} \in R_j^{neg}$, $r_i^{neg} = \{(f_{i1}^{neg}, o_{i1}^{neg}, s_{i1}^{neg}), ..., (f_{ij}^{neg}, o_{ij}^{neg}, s_{ij}^{neg}), ...\}$

The same definitions also apply for the set of positive reviews R_j^{pos}. The neutral orientation reflects the lack of opinion about the specific feature and consequently will not be considered in the reputation model.

Our proposed product reputation model consists of three stages:

- Feature Reputation: the reputation of every feature is calculated based on the frequencies of positive and negative opinions about the features and its sub features.
- Features' Impact: feature impact is used to give a different weight for every feature depending on the number of opinions available in users' reviews about this feature.
- Product Reputation: the final product reputation is the aggregation of features' reputations.

In the following sections we will describe them in details.

3.2 Feature Reputation

The basic idea of the proposed model is to generate the reputation of a product based on the reputation of the product's features. The reputation of each feature is generated based on the opinion orientation and strength of its sub features. For a feature f_i, the reputation of f_i will be the aggregation of the positive and negative opinions weights for all of its sub-features f_j, where $(f_i, f_j) \in L$ as mentioned in Definition 1. This section will discuss how to derive feature reputation based on sub features' opinion information.

Negative Opinion Weight

In this part we suggest a formula to give more weights for frequent negative opinions about a specific feature. By "frequent", we mean that the negative opinion about a feature has occurred in many reviews. Frequent negative opinions may indicate a real drawback in the product, where there is a larger probability that a customer will have the same problem if he buys this product. Thus, when more reviews share a negative opinion about the same feature, the risk of facing the same problem becomes higher. These kinds of problems must appear in the reputation model in order to reflect a true evaluation for the product in use, and to draw user's attention so that he can look for more details and have a rational decision about buying the product. Therefore, we suggest giving these types of negative opinions more weight to draw the user's attention to problems in the products. If we have some negative opinions about different sub-features f_j for the feature f_i, we do not consider them as frequent for the feature f_i. For example, if we have negative opinions about a mobile phone camera as follows "Low video recording quality", "The flash light give a very harsh light", and "No zoom available", these negative opinions about the camera cannot be considered frequent in terms of "camera" because they are about different sub-features (video recording, flash light, and zoom) of the generic feature "camera". Equation (1) is used to calculate the negative weights for each feature f_j.

$$N_j = \sum_{i=1}^{\left|R_j^{neg}\right|} \left(s_{ij}^{neg} + \frac{i + \beta - 1}{\beta} - 1 \right) \tag{1}$$

N_j: is the weight for negative opinions of feature f_j.

$\left|R_j^{neg}\right|$: is the number of reviews that contains negative opinions about the feature f_j.

s_{ij}^{neg} : is the strength of negative opinion in review (*i*) about the feature (*j*).

β : is a positive integer that is used to define the interval of weight increment for the subsequent opinions, where

$$Interval = \frac{1}{\beta} \tag{2}$$

The value of β is subject to change, higher β values will furnish higher feature reputation values, and that is because the $Interval$ value in (2) will be less, which indicates fewer increments in weights for frequent negative opinions. We use ($\beta = 3$); which indicates that the weights for frequent opinions will match with the series in (3), as it appears in the series we keep the value of the opinion strength s_i intact, and we add $Interval$ to increase the weight.

$$N_j = s_{1j}^{neg} + (s_{2j}^{neg} + 0.33) + \ldots + (s_{\left|R_j^{neg}\right| j}^{neg} + \frac{\left|R_j^{neg}\right| - 1}{3}) \tag{3}$$

For a feature f_i which has sub features {f_1, f_2, ..., f_k}, Equation (4) is proposed to calculate the overall weight for negative opinions about the generic feature f_i, which is the sum of the weights of all its sub-features calculated using Equation (1).

$$WN_i = \left(\sum_{j=1}^{k} N_j\right) + N_i \tag{4}$$

WN_i: is the weight of all negative opinions about a generic feature f_i in the hierarchy *FH*.

k : is the number of sub-features of feature f_i .

N_i: represents the weight of negative opinions about the generic feature f_i itself and not about one of its sub-features. It is calculated using Equation (1).

Positive Opinion Weight

For the positive opinions, we propose to calculate the positive weight for a feature f_j by adding opinion strength values s_i given in Equation (5). If the feature has sub features {f_1, f_2, ..., f_k}, the overall weight for positive opinions about the generic feature f_i, is the sum of the positive weights of all its sub-features plus the positive weight of itself, as showed in Equation (6) below:

$$P_j = \sum_{i=1}^{\left|R_j^{pos}\right|} s_{ij}^{pos} \qquad (5)$$

$$WP_i = \left(\sum_{j=1}^{k} P_j\right) + P_i \qquad (6)$$

P_j: is the weight for positive opinions of feature f_j.

WP_i: is the weight of all positive opinions about a generic feature f_i in the hierarchy *FH*.

P_i: represents the weight of positive opinions about the generic feature f_i itself and not about one of its sub-features. It is calculated using Equation (5).

Calculating Feature Reputation

In this paper, we propose to calculate the reputation of a feature based on its overall positive and negative weights as showed in Equation (7), which represents the percentage of positive opinion weights to the total weights of both positive and negative opinions.

$$FREP_i = \frac{WP_i}{WP_i + WN_i} \times 100 \qquad (7)$$

An example is given in Table 1 to demonstrate the proposed method. In the table, for simplicity, each feature listed on the left most column has three sub features; NOf_1, NOf_2, and NOf_3 are the number of reviews which contain negative opinions to the corresponding sub features; N_1, N_2, and N_3 are the negative weight of corresponding sub features; NOF_i and $WNOF_i$ are the number of reviews containing negative opinions about feature F_i and its negative weight respectively. It also shows the total number of positive reviews (*PO*), the total number of negative reviews (*NO*), overall weight for positive (WP_i) and negative (WN_i) opinions, and the aggregation ($FREP_i$) using the proposed method and the (*PPR*) which is the percentage of positive reviews among all reviews without considering the strength of opinion and it can be calculated using Equation (7), where ($WP_i=PO$) and ($WN_i=NO$). (Note: the strength of each opinion was not provided in the table).

The example shows the detailed calculations for both positive and negative opinions weights. In the last two columns we can see the differences between the feature reputation value using our method ($FREP_i$) and the simple average method (*AVG*). Our method results in lower reputation in all cases, this is logical as we give more weight for negative opinions. For example, the total number of negative opinions (*NO*) for both F_2, and F_7 are the same which is equal to 21. Nevertheless, the overall weight for negative opinions (N_2) for F_2 is 63.33 and for F_7 is 53.00, which is totally different. This difference is due to; first, the large frequency for the second sub-feature (NOf_2 = 11) for F_2, second, higher values for opinions' strength (which was not provided in the table). Fig. 1 shows the relation between ($FREP_i$) and (*AVG*) where the difference between the two values is the most when the percentage of negative opinions to positive ones is higher. And this complies with our purpose of giving negative opinions more weight.

3.3 Feature Impact

Depending on the fact that product features are not equally important to customers, we will calculate feature's impact, which is a value that reflects a feature's influence between users. Some of the features are essential for a product to work, but they do not inspire customers to buy the product, as they become consistent over time. On the other hand, some hot features, that are improved continually or new features have high influence on customers to be more interested in the product. Thus, these features should have more impact on the product overall reputation. Features impact will be used to give different weights for every feature in the final product reputation aggregation formula. We suggest that features that frequently occurred in users' reviews have more impact than other features. Let Mj denote the number of reviews that have opinion about this feature, whether positive or negative, the impact of a feature f_j, denoted as $I_{j,}$ is defined in Equation (8) below:

$$M_j = \left|R_j^{neg}\right| + \left|R_j^{pos}\right|$$

$$I_j = \frac{M_j}{M_{Max}} \tag{8}$$

M_{Max}: is the largest value of M_j for all features.

All feature impacts will be given values between 0 and 1; 1 for the feature that was mentioned the most in the users' reviews, and thus has the most influence on users.

Table 1. An example showing the calculation of feature reputation

Features	***NOf***$_1$	***N***$_1$	***NOf***$_2$	***N***$_2$	***NOf***$_3$	***N***$_3$	***NOF***$_i$	***WNOF***$_i$	***PO***	***WP***$_i$	***NO***	***WN***$_i$	***FREP***$_i$	***PPR***
F_1	2	4.33	4	9	1	3	5	15.33	110	266	12	37.67	87.60	90.16
F_2	3	6	**11**	37.33	3	9	4	11	87	170	**21**	**63.33**	72.86	80.56
F_3	10	39.00	9	26	7	22	0	0	215	425	26	87.00	83.01	89.21
F_4	7	19	6	15	3	7	2	4.33	366	722	18	45.33	94.09	95.31
F_5	**13**	49	8	25.33	2	3.33	1	1	145	283	24	78.67	78.25	85.80
F_6	9	30	11	38.33	5	14.33	17	78.33	417	835	**42**	**161.00**	83.84	90.85
F_7	**8**	20.33	5	14.33	3	5	5	13.33	329	655	**21**	**53.00**	92.51	94.00
F_8	12	47	2	6.33	3	6	0	0	273	563	17	59.33	90.47	94.14

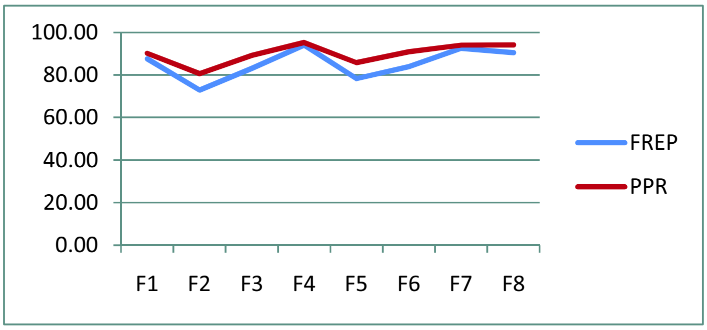

Fig. 1. Feature reputation diagram for the proposed method and the simple average method

3.4 Product Reputation

Many opinions in customers' reviews targeted the product itself rather than mentioning a specific feature in the product, these opinions are also considered in our model. We propose to calculate the product reputation by integrating the reputation calculated based on the reviews which are directly about the product and the reputations of the product's direct features.

Assume that a product has h direct sub features, $FREP_j$ and I_j are the reputation and the impact of each sub feature, respectively. Let GOP denote the product reputation calculated using Equation (7) where WN_i and WP_i are the number of negative and positive opinions about the product itself in the reviews respectively, and GOP have the impact of 1. The following equation is proposed to calculate the product's overall reputation, where every feature reputation, calculated using Equation (7), is multiplied by its impact, calculated using Equation (8), in order to give different weights for features, plus the GOP, and the total is divided by the summation of all features' impacts plus 1 that represents the GOP impact.

$$PR = \frac{\sum_{j=1}^{h}(FREP_j \times I_j) + GOP}{\sum_{j=1}^{h} I_j + 1} \quad (9)$$

Table 2 shows the results of calculating the overall product reputation using our model, and the simple average technique. It shows the values of ($FREP_i$) and (PPR), from Table 1, the (M_j) column indicates how many times this feature and its sub-features have been explicitly mentioned in the reviews, and (I_j) column is calculated using (8) where $M_{Max} = 459$ (the most mentioned feature). It also shows the results of the product reputation (PR) and the regular (AVG).

Table 2. Example Reputation Calculation

Features	*FREP$_j$*	*PPR*	*M$_j$*	*I$_j$*	*FREP$_i$ * I$_j$*
F_1	87.60	90.16	122	0.27	22.90
F_2	72.86	80.56	108	0.24	15.71
F_3	83.01	89.21	241	0.53	41.05
F_4	94.09	95.31	384	0.84	77.06
F_5	78.25	85.80	169	0.37	26.09
F_6	83.84	90.85	**459**	1.00	77.51
F_7	92.51	94.00	350	0.76	68.36
F_8	90.47	94.14	290	0.63	55.05
GOP	**86.31**	**86.31**	**528**	**1.00**	**86.31**
Total	**-**	-	-	5.63	470.03
AVG	**-**	**89.59**	-	-	-
PR	**87.00**	-	-	-	-

As we mentioned before, our model reveals a final reputation lower than the average method. One of the strength points in our model is data representation, as we are able to provide details for customers about every specific feature. Fig. 2 shows the reputation of every feature, which can be more inspiring for customers than the one value reputation representation. Furthermore, more detailed information can also be provided as showed in the example in Fig. 3. For example, if the user is interested in a specific feature and he wants to see more about it, a second level will show the details of negative opinions about sub-features and the frequency of each one.

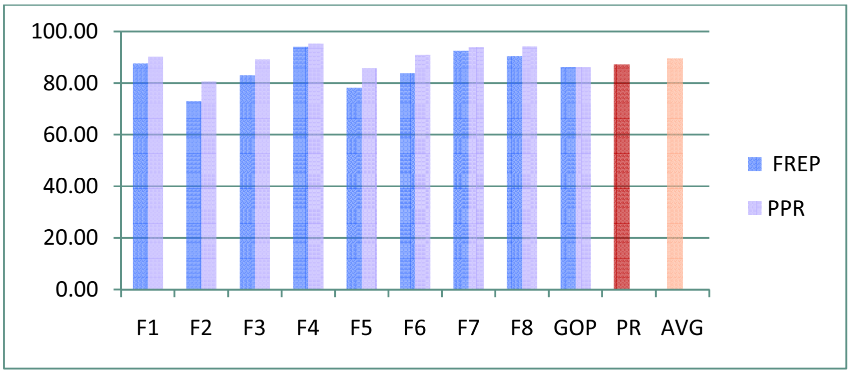

Fig. 2. Results of product reputation model including all features and the regular average result

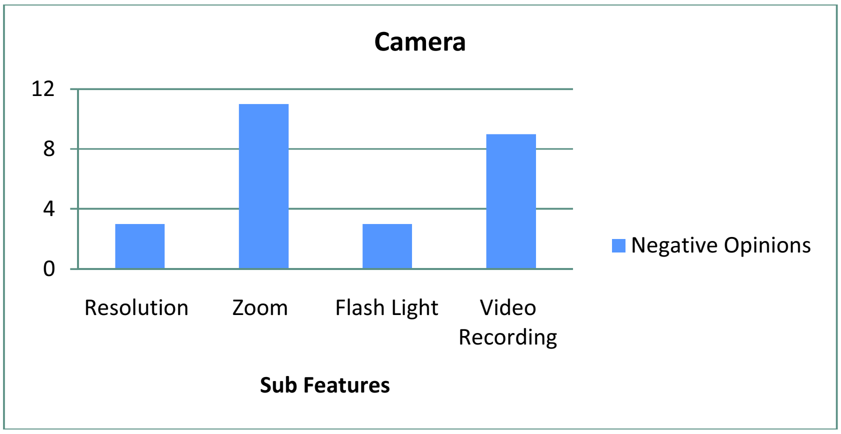

Fig. 3. Example of negative opinions of features at the second level

4 Conclusion

In this paper we have presented a new reputation model for products, our model used text reviews rather than users' ratings. We extracted opinions about hierarchy of features and calculated the frequencies for positive and negative opinions assuming that frequent negative opinions about features and sub-features should get more weight in the reputation calculation, as they indicate a problem in a product a customer may face if they buy it. In Addition, we calculated the impact of features, hence certain features in some products are more inspiring for users, and therefore they are more important in the reputation model. Our model integrates the strength of opinions and provides summary about users' opinions for customers rather than representing reputation as a number of stars. For future work, the reputation model may be modified to consider age and validity of reviews, and also detect malicious users' reviews which aim to sabotage the reputation of a product.

References

1. F. Garcin, B. Faltings, and R. Jurca, "Aggregating reputation feedback," in *International Conference on Reputation ICORE*, 2009, pp. 119-128.
2. P. Resnick, K. Kuwabara, R. Zeckhauser, and E. Friedman, "Reputation systems," *Communications of the ACM (CACM),* vol. 43, pp. 45-48, 2000.
3. J. Cho, K. Kwon, and Y. Park, "Q-rater: A collaborative reputation system based on source credibility theory," *Expert Systems with Applications,* vol. 36, pp. 3751-3760, 2009.
4. S. Leberknight, S. Sen, and M. Chiang, "On the Volatility of Online Ratings: An Empirical Study " in *E-Life: Web-Enabled Convergence of Commerce, Work, and Social Life*. vol. 108, ed: Springer Berlin Heidelberg, 2012, pp. 77-86.
5. B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Foundations and Trends in Information Retrieval*, vol. 2, pp. 1-135, 2008.

6. M. Hu and B. Liu, "Mining and summarizing customer reviews," in *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2004, pp. 168-17
7. J. Yi and W. Niblack, "Sentiment Mining in WebFountain," in *IEEE International Conference on Data Engineering (ICDE)*, 2005, pp. 1073-1083.
8. P. D. Turney, "Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews," in *Association for Computational Linguistics Conference (ACL)*, 2002, pp. 417-424.
9. B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," in *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2002, pp. 79-86.
10. J. Kamps, M. Marx, R. J. Mokken, and M. De Rijke, "Using wordnet to measure semantic orientations of adjectives," in *International Conference on Language Resources and Evaluation (LREC)* 2004, pp. 1115-1118.
11. A. M. Popescu and O. Etzioni, "Extracting product features and opinions from reviews," in *Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT/EMNLP)*, 2005, pp. 339-346.
12. S. Morinaga, K. Yamanishi, K. Tateishi, and T. Fukushima, "Mining product reputations on the Web," in *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2002, pp. 341-349.
13. T. Hashimoto and Y. Shirota, "Semantics extraction from social computing: a framework of reputation analysis on buzz marketing sites," in *International Conference on Databases in Networked Information Systems (DNIS)*, 2010, pp. 244-255.
14. S. Baccianella, A. Esuli, and F. Sebastiani, "Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining," in *International Conference on Language Resources and Evaluation (LREC)*, 2010, pp. 2200-2204.