A Feature-Based Reputation Model for Product Evaluation

Umar Farooq[*,‡,§,¶], Antoine Nongaillard[†,‖], Yacine Ouzrout[‡,**]
and Muhammad Abdul Qadir[§,††]

[*]*Department of Computer Science*
Abdul Wali Khan University Mardan, Pakistan

[†]*CRIStAL Laboratory, CNRS UMR 9189*
Lille University, Cité Scientifique
Villeneuve d'Ascq, Lille, 59650, France

[‡]*DISP Laboratory, University Lumiere Lyon 2*
160 bd de l'Université, Bron cedex
Lyon, Rhne-Alpes, 69676, France

[§]*Department of Computer Science*
Capital University of Science and Technology
Islamabad, 44000, Pakistan
[¶]*Umar.Farooq@univ-lyon2.fr*
[‖]*antoine.nongaillard@univ-lille1.fr*
[**]*yacine.ouzrout@univ-lyon2.fr*
[††]*aqadir@cust.edu.pk*

Knowing the strengths and weaknesses of a product is very important for manufacturers and customers to make decisions. Several sentiment analysis systems are proposed to determine the opinions of customers about products and product features. However, the aggregation methods used are not able to estimate a true reputation value and to reflect the recent opinions quickly. Most of these systems are based on single source and therefore suffer from availability and susceptibility issues. In this paper, we propose a multi-source reputation model where several aggregation methods are introduced in order to evaluate product based on features. In addition, we also propose a method which uses four parameters in order to rank the reputability of each rating before considering it for reputation values. The results show that the proposed model estimates good reputation values even in the presence of biased behaviors, robust to false ratings and reflects the newest opinions about product rapidly.

Keywords: Product reputation model; product evaluation; reputation system; feature reputation; ratings aggregation.

1. Introduction

The popularity of online rating systems cannot be denied nowadays. Blogs, discussions forums, e-commerce and review sites give opportunities to customers to post reviews in order to express their opinions about products and services. This process

[**]Corresponding author.

leads to the generation of a huge amount of product evaluation data which can be used by customers and companies in order to make decisions. Indeed, when a customer want to buy a product (or use a service), his/her first idea is to check the opinions of other customers about this product on reviews and e-commerce sites. The information available on these ratings sites is used by customers to check the reputation and to compare different products. Thus, e-commerce and review sites can influence customers in the selection of the most suitable product. Similarly, companies can also use the product evaluation data either to position a new product in a market or to improve existing products.

In order to exploit product evaluation data, different reputation models have been proposed in the literature. These models can be classified based on the type of ratings aggregated or analyzed. On reviews and e-commerce sites users can express their opinions in the form of *numeric ratings* (also called star rating) or *textual descriptions* (called review). For instance, on e-commerce and review sites such as *Amazon*,[a] *Ebay*,[b] *Cnet*[c] and *Epinion*[d] both numeric and textual ratings are posted. However, these sites only aggregate numeric ratings using simple arithmetic mean to determine a "star value" which represents the product reputation. Other aggregation methods such as weighted mean, median or mode are also used in the literature to aggregate numeric ratings. The exclusive consideration of numeric ratings makes reputation a general subjective measure where different aspects of a product cannot be distinguished in the aggregated evaluation. Furthermore, an aggregated star value (ASV) does not provide enough information about a product to make decisions. Indeed, customers have different profiles and may be interested in different aspects of a product.

Other studies on sentiment analysis which are either supervised or unsupervised are also proposed in the literature. These methods analyzed textual ratings in order to produce a summary of opinions about a product or a specific feature of a product.[1–7] Such studies are based on text mining and natural language processing techniques, which are used to determine whether opinions about product (or a product feature) are positive or negative. However, results of sentiment analysis are aggregated using simple summation most of the time. Some other aggregation methods such as "*weighted mean*", "*ratio of positive over all opinions* (ROPOAO)"[8] or "*difference of positive and negative over all opinions*"[9] have also been proposed. Results obtained using these aggregation methods can be easily biased using few false ratings. Moreover, these methods based on sentiment analysis do not reflect quickly drastic change in reputation since the same weight is assigned to recent and old ratings. More information should be considered when a reputation value is computed. Indeed, the rating trustworthiness, the reliability of a source and the feature

[a] http://www.amazon.com/.
[b] http://www.ebay.com/.
[c] http://www.cnet.com/.
[d] http://www.epinion.com/.

looked by a customer or the expertise level of the reviewer can also be useful information.

In most of existing product reputation models, only one kind of rating that is either numeric or textural rating is considered while computing reputation value. Online rating systems only aggregate numeric ratings while reputation systems based on sentiment analysis only analyzed textual ratings. Considering both numeric and textual ratings allow us to compute several reputation values and increase the choices for customers and companies to evaluate products in different ways. In addition, the existing models also ignore other evaluation information such as review date, review title, review helpfulness, etc. which can be used to compute different rating parameters.

Most of existing reputation models compute reputation value based on the ratings extracted from a single web source. Single source approaches are much more sensitive to false information.[10] In addition, sometimes single source lacks ratings in order to compute a good reputation value and may suffer from availability issue because the web source may not be available.

In order to address the issues discussed, in this paper we propose a multi-source product reputation model where several aggregation methods are introduced in order to evaluate product based on features. These aggregation methods provide a good estimation of the actual reputation values even if some ratings are false or biased. These aggregation methods are robust to false ratings and are also able to reflect recent opinions about product quickly. The proposed model allows the users to compute reputation values based on different criteria. It considers four parameters namely *source reliability*, *reviewer expertise*, *ratings trustworthiness* and *ageing factor* to determine the reputability of rating, which improves the reliability and the accuracy of the reputation value.

This paper is organized as follows. In Sec. 2, related works are presented and analyzed. Section 3 gives an overview of the proposed model. Section 4 explains different rating parameters on the bases of which rating reputation is computed. Section 5 describes different aggregation methods proposed to compute feature and other reputation values. Finally, Sec. 6 describes experimental settings and results.

2. Related Work

This section has been organized as follows. First, existing product reputation models and aggregation methods are described. Second, the performance of existing reputation models and aggregation methods are analyzed against six constraints which an aggregation method should satisfy. Last, the issues in existing aggregation methods are summarized into a tabular form.

Reputation models have been investigated by many authors during the last decade. Many studies are dedicated to computing user trust and reputation in e-commerce environment, while very few of them focused on product reputation.[1–9] Product reputation models and aggregation methods can be classified according to

the type of ratings considered; i.e., numeric ratings and textual ratings. Several online product rating systems are available which only aggregate numeric ratings using simple arithmetic mean to determine ASV. However, arithmetic mean is not a suitable aggregation method because it is not informative, strategy proof and robust to false ratings.[11] Some other aggregation methods such as mode and median are also proposed for aggregation numeric ratings which are robust to false ratings.[12] However, the median is the only strategy proof aggregation method.[13] These statements are validated with formal justification later while discussing different constraints. The issue with both mode and median is that they aggregate similar kind of information regardless the importance of each observation or rating. In product reputation, some ratings may be more important than others and hence need to be given more weight over less important ratings. Through classical mode and median, we are not able to aggregate information such as rating trustworthiness, ageing factor and reviewer expertise while computing reputation value. Mode and median also need tie breaking rule when more than one results are possible. However, optimal tie breaking rule is difficult to establish especially when the difference between possible results is high. On the other hand, the issue concerned with numeric ratings is that it is a general subjective measure[14] which is given on the bases of average performance of the products. Sometimes users give ratings on the bases of single aspect which is more important for them and hence in such situations, this measure is difficult to protect from unfair ratings. In addition, the ASV computed based on numeric ratings does not provide enough information to customers and manufactures in order to make decisions. In this paper, we are not focusing on numeric ratings but on the aggregation of sentiment analysis results of textual ratings to compute feature and other reputation values so that enough information for decision making can be provided.

On the other hand, the reputation models based on opinion mining analyzed textual ratings in order to form a summary of opinions either about product and/or product features.[3,5–7,15] Some aggregation methods are also proposed which used the results of opinion mining in order to compute reputation value. These aggregation methods can be further categorized based on the reputation value computed. Two types of reputation values, i.e., feature reputation and feature-based product reputation values are computed in the literature. In Ref. 3, association mining is used to identify product features from customer reviews. The synonym and antonym sets of WordNet[e] are utilized to infer the sentiment orientation of adjectives in order to summarize the opinions about product features. In another study, an unsupervised classification method relaxation labeling is proposed to determine the opinions about product features.[5] Morinaga *et al.*[6] used four data mining techniques: rule analysis, co-occurrence analysis, typical sentence analysis and corresponding analysis to mine features opinions. The position map is used to visualize the results. In all these methods based on opinion mining, the researchers focused on sentiment analysis instead of aggregation methods and therefore simple summation is used as an

[e] https://wordnet.princeton.edu/.

aggregation method to compute feature reputation. In Ref. 8, a mathematical model based on opinion mining is proposed. The reputation of a product feature is computed using the "RPOAO". The important features are given more weight while computing feature base product reputation value. However, the feature base product reputation value is computed using simple weighted mean of all feature reputation values, which has several issues.[11,12] We called this aggregation method as W.Mean +RPOAO because first RPOAO is used to compute features reputation and then weighted means (of those feature reputation values of all features of a product) is used to compute feature-based product reputation value. A manual hierarchy of the product features and subfeatures is developed and positive opinions about subfeatures are also considered while determining a feature reputation. However, developing a manual hierarchy for every product is difficult and time consuming. Furthermore, no proper experiment is performed to validate the results. Another study[9] proposed a product review classification method where reputation associated with a feature is computed using the difference between the total number of positive and negative reviews in which a feature appeared divided by the total number of reviews. We called this aggregation method as difference of positive and negative over all opinions (DPNOAO). Instead of only considering the opinions related to a specific feature, this method computes feature reputation based on the overall opinion expressed in the reviews where the feature appeared. A positive classified review contains more positive opinions but it may also contain negative opinions about some features. However, these negative opinions appeared in a positive classified reviews are also considered positive while determining feature reputation.

In order to analyze, the performance of existing aggregation methods, we define six formal constraints. An aggregation method should satisfy these constraints in order to estimate a good reputation value in different scenarios. Justifications have also been provided which explains the reasons that why a particular constraint need to be satisfied.

(i) An aggregation method must be robust to false and biased ratings. Robustness measures the extent to which the estimation of an aggregation method is unaffected by outliers or malicious reviews. We define the breakdown point in order to measure that how robust the existing aggregation methods are. The breakdown point is the proportion of outliers or false ratings required to make the aggregation method return to an arbitrary value. Let $\{r_1, r_2, \ldots, r_{n-h}, r_1^{'}, r_2^{'} \ldots, r_h^{'}\}$ is a sample of n numeric ratings where r_i are genuine and $r_i^{'}$ are false or biased ratings. The breakdown point ϵ of an aggregation method $\bar{r_a}$ is the smallest proportion $\frac{h}{n}$ for which the false ratings $\{r_1^{'}, r_2^{'} \ldots, r_h^{'}\}$ will cause the reputation value to change. An aggregation method will be more robust if the breakpoint is higher.

The simple arithmetic mean can be defined by Eq. (2.1). One false or biased rating is enough to change the reputation value by arithmetic mean and hence the breakpoint is equal to $\epsilon = \frac{1}{n}$. Similarly, when ratings are unbounded and we

want to significantly change the arithmetic mean from $\bar{r}_{mi}$ to an arbitrary value $\bar{r}_{mj}$, we may need more than one false ratings V with arithmetic mean $\bar{r}'_{mv}$ as defined by Eq. (2.2). For example, to bring the arithmetic mean from 4.5 (with $n = 100$) to 4, we need $V \geq 17$ false or biased ratings with one star.

$$\bar{r}_m = \frac{1}{n}\left(\sum_{i=1}^{n-h} r_i + \sum_{i=1}^{h} r'_i\right), \tag{2.1}$$

$$V \geq \frac{n(\bar{r}_{mi} - \bar{r}_{mj})}{(\bar{r}_{mi} - \bar{r}'_{mv})}. \tag{2.2}$$

The weighted mean is similar to simple arithmetic mean except the weights associated with each rating. The weight represents the importance of each rating based on some criteria. The weighted mean can be define by Eq. (2.3). The breakdown point does not change with weights in weighted mean and remains the same as in arithmetic mean. Similarly, the number of false ratings needed to change the weighted mean is upper bounded by Eq. (2.2) and it also depends on the weights associated with ratings. Therefore, both mean and weighted mean are not robust to false and biased ratings.

$$\bar{r}_{wm} = \frac{\sum_{i=1}^{h} w_i.r'_i + \sum_{i=1}^{n-h} w_{(i+h)}.r_i}{\sum_{i=1}^{n} w_i}. \tag{2.3}$$

The median can be define as the rating $\bar{r_e}$ which separates the higher half of the ratings sample from the lower half. Let $\{r_1, r_2, \ldots, r_{n-h}, \acute{r_1}, \acute{r_2} \ldots, \acute{r_h}\}$ be the set of ratings for a product g arranged in ascending order. If the total number of ratings n is odd (i.e., $n = 2k + 1$), then the median is positioned at location $(k + 1)/n$. On the other hand, if n is even that is $n = 2k$, then the median is at k. In order to measure the robustness, we determine the breakdown point by using Eq. (2.4). Similarly, in order to determine the number of false ratings needed to change the medium significantly we need at least $n + 1$ ratings in worst case. For example, if we want to change the medium of product g with 100 ratings, then we need at least 101 false or biased ratings. This shows that medium is the robust aggregation method to false and biased ratings.

$$\epsilon = \begin{cases} \dfrac{k+1}{n} = \dfrac{1}{2} + \dfrac{1}{2n}, & n = 2k + 1, \\ \dfrac{k}{n} = \dfrac{1}{2}, & n = 2k. \end{cases} \tag{2.4}$$

The mode $\bar{r_d}$ is the rating which appears most frequent in ratings set. Let x and y are the number of ratings with the star value r_1 and r_2 respectively and $x \neq y$. Indeed, if the mode is rating $\bar{r_1}$ then $x \geq y + 1$. Therefore, the breakdown point of the mode is give by $\epsilon = ((n/2) + 1)/n$ for n ratings. Based on this reasoning,

the number of false ratings needed is equal to $V = n + 1$ and hence we can say that mode is also a robust aggregation method.

(ii) An aggregation method should be strategy proof. A strategy proof aggregation method does not provide incentive to a reviewer in order to obtain their preferred reputation value by lying or hiding the actual evaluation. A rater has their preferred score for a product that he wants to see as an aggregated value. When the aggregated value is different from the preferred score, then a reviewer may post rating that differ the actual perception in order to bring the aggregated score equal or even closer to the preferred score. The mean and weighted mean are not strategy proof because any reviewer has the incentive to rate differently in order to push the mean or weighted mean value downwards or upwards. Similarly, the mode is also not a strategy proof aggregation method. Assume that three reviewers have posted ratings with star value r_1 and four reviewers have reported ratings with star value $r_2 > r_1$. Let a_v is a reviewer whose true perception is $r_j < r_1$. However, he can misreport by rating the product with the star value r_1 in order to successfully change the mode of the product from r_2 to r_1, which is more desirable outcome for a_v. Moulin[13] proved that the median is the only strategy proof aggregation method when preferences are single peaked along the real line. Assume that a product g has the rating set $\{r_1, r_2, \ldots, r_n\}$ sorted in increasing order having median $\bar{r_i}$. Misreporting a lower $r_j < r_i$ or higher value $r_k > r_i$ by a reviewer most likely will not change the median and less probably can either increase or decrease the median further along the real line.

(iii) An aggregation method must consider the rating trustworthiness while determining the reputation value. Let $\{r_1, r_2, \ldots, r_{n-h}, r_1^{'}, r_2^{'}, \ldots, r_h^{'}\}$ be a sample of ratings for a product g which includes both genuine ratings r_i and false or biased ratings $r_h^{'}$. Indeed, if an aggregation method $\overline{r_a}$ considers both genuine and false ratings without any differentiation then the reputation value can be misled. Several methods are proposed in the literature in order to determine spam reviews.[16–18] Some of these methods provide results in binary classification which can be easily integrated with all aggregation methods.[18] On the other hand, some sophisticated methods also produce results in probability estimation which actually represent the likelihood that a review is spam.[16,17] However, most of these aggregation methods such as mean, mode, median, RPOAO and DPNOAO are not able to aggregate such kind of information. As an alternative to rating trustworthiness, some reputation models used robust aggregation methods such as median and mode which are less affected by outliers. However, these aggregation methods can only estimate a good reputation value when the outliers are less. In the case when the outliers are more than 50%, then the reputation value will almost be completely manipulated.

(iv) An aggregation method must be able to compute the recent reputation value. Suppose that a product with a ratings set $\{rh_1, rh_2, \ldots, rh_c\}$ has good

reputation and all the reviewers rate the product with highest star value rh_i. However, due to some reasons the quality become worst and the users started to rate the product with lowest star rl_i and hence more ratings have been added to the ratings sample $\{rh_1, rh_2, \ldots, rh_c, rl_1, rl_2, \ldots, rl_e\}$, where c is the total number of old ratings (with highest star) and e is the total number of recent ratings (with lowest star). If an aggregation method $\overline{r_a}$ does not favor the recent ratings rl_i, then definitely the reputation value can be misled. The existing aggregation methods are not able to reflect the recent ratings because the same weight is given to recent and old ratings. Therefore, they also allow the manufacturer to maintain the good reputation for a long time even if the quality of the product become worst. In order to address this issue, the recent ratings need to be given more weight over old ratings, which can be achieved by using an ageing factor. Let the weights $\{w_1, w_2, \ldots, w_{n'}\}$ given by a decay principle to ratings from recent to old such that $w_1 \geq w_2 \cdots \geq w_n$. The weights should be given in such a way that they satisfy the following two sub constraints. First, as there is a trade-off between sensitivity and robustness,[19] therefore favoring recent ratings must not compromise the robustness of an aggregation method. Second, the weights should be distributed fairly to cover enough samples of ratings in order to estimate a good reputation value. The aggregation method will behave differently to false and genuine ratings if constraints (i), (ii) and (iv) are satisfied. These constraints will make the aggregation method robust to false and biased ratings but sensitive to genuine ratings, which is more desirable. However, it should not be too much sensitive even to genuine ratings so that the malicious reviewers who can bypass the fake review detection method cannot easily manipulate the reputation value.

(v) The ratings from unreliable sources should be discounted. Suppose that a reputation model extracts ratings from multi web sources. Let the sources set is $\{s_1, s_2, \ldots, s_n\}$ then all the sources may not be reliable because some source may have incentives as a middle man or seller and may promote certain products. Therefore, the source reliability SR_i for all sources $\{\mathrm{SR}_1, \mathrm{SR}_2, \ldots, \mathrm{SR}_n\}$ need to be determined and considered while computing the reputation value in order to increase the reliability and accuracy of reputation value. However, the existing aggregation methods do not consider source reliability while computing reputation value and hence vulnerable to malicious web sources.

(vi) An aggregation method should compute the reputation value with good accuracy. Suppose we have a rating set $\{r_1, r_2, \ldots, r_{n-(h+k)}, r_1', r_2', \ldots, r_h', \hat{r_1}, \hat{r_2}, \ldots, \hat{r_k}\}$ where r_i are the genuine ratings, r_i' are the false or biased ratings and $\hat{r_i}$ are the ratings from a malicious web source. We assume that the actual reputation R_a is based on genuine ratings extracted from trustworthy web source and also based on recent ratings. Let R_p is the reputation value computed by an aggregation method $\bar{r_m}$ and the absolute different between R_a and R_p is called absolute error (AE) $\mathrm{AE} = |R_a - R_p|$. An aggregation method is said to be accurate if AE is either equal or very close to R_p. All the constraints

Table 1. Issues in existing aggregation methods.

Aggregation method	Robustness	Sensitivity	Balance b/w R and S	Strategy proofness	Similar kind of info	Tie breaking rules
Mean[11,12]	No	No	No	No	Yes	No
W.Mean[11,12]	No	No	No	No	No	No
Mode[11,12]	Yes	No	No	No	Yes	Yes
Median[11–13]	Yes	No	No	Yes	Yes	Yes
RPOAO[8]	Yes	No	No	Yes	Yes	No
DPNOAO[9]	Yes	No	No	Yes	Yes	No
W.Mean+RPOAO[8]	Yes	No	No	No	No	No

defined so for contribute to the accuracy of an aggregation method. As the existing aggregation methods do not satisfy most of these constraints hence they may not be able to provide a good estimation of the actual reputation value.

The issues in existing aggregation methods (i.e., for both numeric and textual ratings) which are discussed throughout this section are summarized in Table 1. Six evaluation criteria which are robustness, sensitivity, balance between robustness and sensitivity (Balance b/w R and S) strategy-proofness, aggregation of similar kind of information regardless of the importance of each rating (similar kind of info) and tie breaking rules are used to evaluate the performance of different aggregation methods.

3. Overview of the Proposed Model

Based on the need of customers and organization for a product reputation evaluation approach, we propose in this section a global architecture answering the research issues highlighted in the literature review. The overall process of the proposed reputation model is divided into four phases, as illustrated in Fig. 1. This figure also shows that how different phases are related to each other and which evaluation information is used in order to compute rating parameters, rating reputation and then reputation values. During first phase, product reviews are extracted from e-commerce and review sites. In second phase, textual ratings are analyzed using sentiment analysis to identify features and to summarize opinions about product and product features. In third phase, first rating parameters such as reviewer expertise, source reliability, ageing factor and rating trustworthiness are computed and then aggregated to compute rating reputation. In last phase, aggregation methods are used to compute different reputation values such as "feature reputation", "feature based product reputation", "aggregating general opinions", "product reputation" and "product reputation based on review titles". The remaining part of this section explains the first two phases and Sec. 4 discusses phase three while Sec. 5 is dedicated to the last phase.

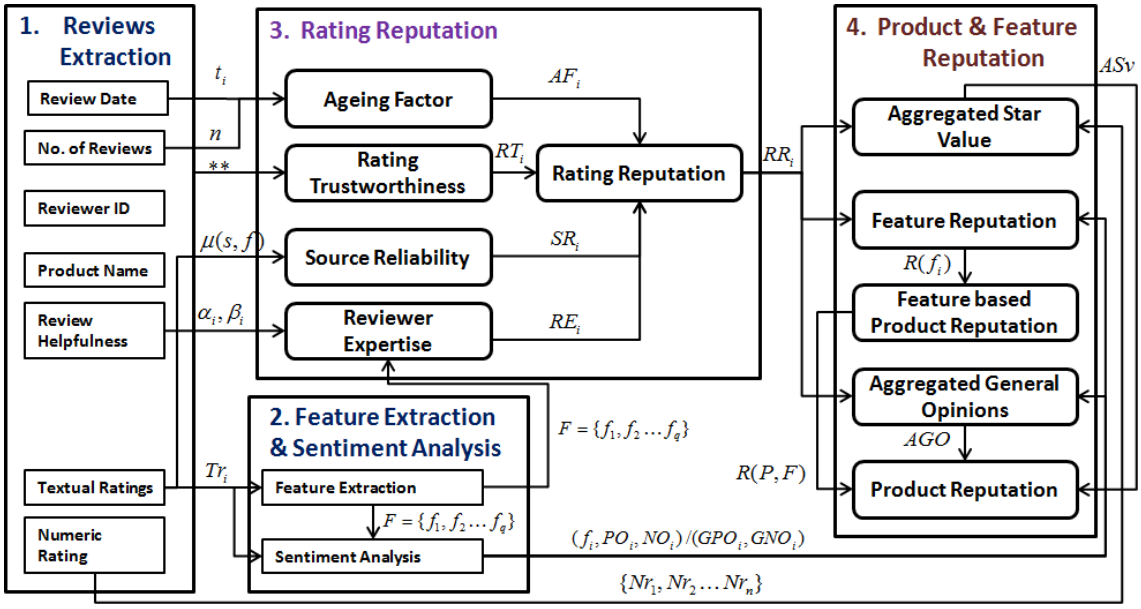

Fig. 1. Overview of product reputation model. Each arrow is labeled with the information that is flowing between different phases, where t is the review date, $\mu(s,f)$ is number of opinions that source s has about feature f, α is positive helpfulness votes, β is negative helpfulness votes, Tr is textual ratings, Nr is numeric rating, AF is ageing factor, RT is rating trustworthiness, SR is source reliability, RE is reviewer expertise, F is product features set, PO is positive opinion, NO is negative opinion, GPO is general positive opinion, RR is rating reputation, ASv is aggregated star value, $R(f)$ is feature reputation, $R(P,F)$ is feature based product reputation value and AGO is aggregated general opinions, **: rating trustworthiness uses most of the information shown in reviews extraction phase.

3.1. *Reviews extraction*

Many websites allow the customers to post reviews in order evaluate the products according to their own experience. In order to extract reviews from these e-commerce and review sites such as *Ebay*, *Cnet* and *Amazon*, we developed a wrapper. The wrapper is fully automatic as the users simply need to select a product in order to extract reviews about that product. The wrapper automatically locates the web pages which contain reviews about the selected product and extracts reviews from all those web pages. The reviews are extracted whenever the reputation is needed to be computed and hence the reputation value is based on up to date ratings. These review sites are selected because they allow the users to post reviews about versatile categories of products. They are also popular among users and hence have sufficient number of reviews about most of the products. In addition, these sites not only make public the textual comments but also other reputation information (such as reviewer id, date on which review is posted, reviews helpfulness votes, etc.) which are used by our model in order to evaluate product based on different criteria. The review or e-commerce site from which reviews are extracted is called web source or simply source, represented by s and belongs to a set of sources $S = \{s_1, s_2, s_3, \dots, s_d\}$. The total number of reviews extracted from a source is represented by $m(s)$ and the total number of reviews extracted from all sources is represented by n. The reviews

extracted from these sites contain information such as product name, author name, date on which review is posted, numeric rating, review title, textual rating and the number of people who found the review helpful. Generally, the feedback in textual form is considered as a review. However, we consider that a review r is a tuple of eight elements $\langle g, \mathrm{Nr}, \mathrm{Tr}, \mathrm{rt}, t, a, \alpha, \beta \rangle$ and belongs to a set of n reviews $R = \{r_1, r_2, r_3, \ldots, r_n\}$. Where g is the product name, Nr is the numeric ratings, Tr is textual rating, rt is the review title, t is the review date, a is the author id, α and β are the positive and negative review helpful votes, respectively. The numeric and textual ratings are the two ways to evaluate the product, which can be used to compute reputation values. The review date, author id, review title and review helpfulness are also very important to determine rating parameters such as rating trustworthiness, reviewer expertise and ageing factor. Unfortunately, the existing product reputation models considered either only numeric ratings or textual ratings to compute reputation and ignored other evaluation information. Our model considers all these evaluation information as shown in Fig. 1 in order to compute different rating parameters and reputation values.

3.1.1. *Numeric rating*

Numeric rating, also called *star rating*, is usually scaled from one to five stars: $\mathrm{Nr} \in \{1, 2, 3, 4, 5\}$. This is a general-subjective measure which reflects the average performance of a product considering all its aspects.[14] Usually, the users give one or two star to the product with lower quality, three to the product with average quality and higher star to the product with a comprehensive range of features, benefits and quality. However, sometimes this measure is difficult to protect from unfair ratings. For example, a user may give lowest numeric rating only based on a single feature which is more important for them, even though the other features perform well. Similarly, instead of giving numeric rating based on average performance of the product a user may give this measure on the bases of services associated with the product such as maintenance service, delivery service or intermediate services, etc.

3.1.2. *Textual rating*

The textual rating is a specific-objective measure,[14] which is issued after assessing an aspect of a product against some criteria. For example, a user may post textual comments such as "the mobile is reliable" and "the processor speed is good". In these comments, the user given their opinions after assessing mobile and processor against reliability and speed criteria, respectively. In textual ratings Tr, user expresses their opinion O about product g and different features F of the product, where F is a set of features $F = \{f_1, f_2, f_3, \ldots, f_q\}$. The opinion O about a feature f can be positive, negative or neutral. The positive and negative opinions are represented by PO and NO, respectively. The total number of opinions about a feature f is represented by $\mu(f)$. The textual ratings are very important because they contain evaluations about different features of the product which cannot be obtained from numeric ratings.

However, textual ratings must be first processed and analyzed in order to determine different features, corresponding opinions and to produce a summary of opinions in some numerical form so that we can able to compute feature reputation and other reputation values.

3.1.3. *Review date*

Review date t is an important aspect especially for companies when it comes to study the evolution of a product reputation, identifying trend in the market and determining the recent reputation. For example, it looks very attractive for both customer and manufacturer to examine the life time of either product or a specific feature reputation. The reputation of product or feature can be determined for historical study by considering the ratings posted in a specific duration. On the other hand, opinions about a product change with the passage of time due to technology improvement, change in quality or trends in the market. Therefore, recent ratings are more important than older ones in order to reflect recent reputation. The review date is integrated in our model through the ageing factor in order to reflect the recent opinions or to perform historical study of the product reputation.

3.1.4. *Review helpfulness*

Review and e-commerce sites allow users to vote for the helpfulness of reviews. This allows the users who read the review to post his vote that whether the review contents are useful for customer or not. The system tracks the total number of positive helpfulness votes α and negative helpfulness votes β about a review and displays it aside a review. This measure is used in our model to estimate the expertise of reviewer.

3.1.5. *Review title*

The review title, rt, is also very important in context of reputation, because usually the users provide a summary about the review in the title. For example, a user may summarize the review in title by using words such as *Excellent Technology*, *Terrible battery* or *Amazing Features*. On the other hand, sometimes the users simply provide the name of the product instead of summarizing the review. The review title is also textual description and hence need to be analyzed to convert it into numeric form so that we can compute *Product Reputation based on Review Titles* value.

3.1.6. *Other evaluation information*

Some other evaluation information such as author id, product name, number of reviews, reviewer age, sex and location are also very useful. Author identification which identifies each reviewer is useful to detect duplicate spam reviews from the same author. In some sources, the author identification is the actual customer name who bought the product, however in most sources the author id can be any name that user select for his identification. Some sources also provide information about

reviewers such as age, sex and location on separate web page which can be used to evaluate product on different criteria. Some other reputation information which are not actually the part of review and available on review page are also helpful while extracting these evaluation information such as page title, total numbers of reviews and total web pages of reviews about the same product.

3.2. *Feature extraction and sentiment analysis*

Sentiment analysis is an automatic way to analyze textual reviews in order to determine that whether the opinions about a product/product feature is positive or negative. In order to determine opinions about product features, first we need to identify all those features about which opinions are expressed in customer reviews. For this purpose, the method proposed in Ref. 3 is implemented which is based on the fact that product features are usually nouns or noun phrases. The Apache OpenNLP part of speech tagger[f] is used to tag each word in reviews with their associated part of speech. The frequent features are identified by determining the frequent nouns and noun phrases in all reviews about a product using association mining. It is more likely that frequent nouns and noun phrases are product features, however some frequent nouns and noun phrases may not be product features. Therefore, an interface is built where the users can remove all those nouns and noun phrases which are not product features and can also select the product features which are not frequent.

After identifying product features the sentences which contain opinions about a product or different features are identified and sentiment analysis method proposed in our previous study[20] is used to determine that whether the opinion is positive or negative. In this method, the opinion bearing words (i.e., adjectives, verbs, adverbs and nouns) are identified using part of speech tagger and the polarity scores of these words are obtained from the SentiWordNet[g] which is a lexicon dictionary of opinionated words. The polarities of words in a sentence are aggregated by pursuing the following steps which are defined in a previous research work.[20] (a) If a negation word is found in a sentence, then the sequence of words affected by the negation is determined and the polarities of these words are inverted before aggregation. (b) When the sentence is simple (which contains one clause) then words polarities are simply added to determine the sentence polarity. (c) On the other hand, if the sentence is compound (i.e., containing multiple clauses), then first the polarities of clauses are determined by adding all worlds' polarities and then the sentence polarity is determined by aggregating all clauses' polarities according to the conjunction rules proposed in Ref. 20.

The sentiment analysis method gives result in binary form, which represents the presence or absence of positive and negative opinions. For example, if a sentence containing opinion about a feature is classified positive, then the result is represented as PO = 1 and NO = 0. The results of sentiment analysis about all opinions either

[f]http://opennlp.apache.org/.
[g]http://sentiwordnet.isti.cnr.it/.

about product or about different features of the product are later aggregated while considering ratings parameters in order to determine reputation values such as feature reputation and aggregating general opinions value.

4. Rating Reputation

A unique characteristic of our reputation model is that it first computes the reputation of each rating before considering it for computing reputation values. The reputation of each rating is determined by considering aspects such as whether the rating is fake or genuine, the expertise of reviewer who posted the rating, the reliability of the source from where the rating is extracted and the time on which rating is posted. Rating reputation measures the extent to which a rating is reputable to be considered for computing product and feature reputation. Rating reputation ranks each rating between one and zero. This value is based on four rating parameters (i.e., source reliability, reviewer expertise, rating trustworthiness and ageing factor).

We proposed Eq. (4.1) to compute rating reputation RR for rating c, where SR is the source reliability of the source from which rating c is extracted, RE_c is expertise of reviewer who posted c, RT_c is the rating trustworthiness, AF_c is the time weight assigned to c using ageing factor and n is the total number of ratings.

$$\mathrm{RR}_c = \frac{\mathrm{SR}_c \times \mathrm{RE}_c \times \mathrm{RT}_c \times \mathrm{AF}_c}{\sum_{i=1}^{n} (\mathrm{SR}_i \times \mathrm{RE}_i \times \mathrm{RT}_i \times \mathrm{AF}_i)}. \tag{4.1}$$

4.1. *Source reliability*

Source reliability measures the extent to which the ratings of a source are suitable to be considered for computing reputation. Source reliability depends on the number of opinions a source has about a feature. The source which has more opinions about a feature is given more weight over the source which has fewer opinions about that feature. Three different methods are used to compute source reliability, each of which is used to compute rating reputation for different reputation values. The source reliability for computing single feature reputation only depends on the number of opinions that each source has about that particular feature. Equation (4.2) is proposed to compute source reliability $\mathrm{SR}(s, f, g)$ of source s for a single feature f of product g.

$$\mathrm{SR}(s, f, g) = \frac{\mu(s, f, g)}{\mu(\max, f, g)} \div \sum_{i=1}^{d} \left(\frac{\mu(s_i, f, g)}{\mu(\max, f, g)} \right), \tag{4.2}$$

where $\mu(s, f, g)$ is the number of opinions that s has about a feature f of product g, $\mu(\max, f, g)$ is the maximum number of opinions that a source has about the same feature of the product and d is the total number of sources. A threshold value of source reliability can be used to ignore the ratings of sources which have very few number of opinions about a feature because a source with only few number of reviews about a product are more likely to be spam.[16] The threshold value can also be useful

when the user want to compute reputation value only based on the ratings posted on web sources which have high source reliability. In this way, the accuracy and reliability of reputation value can be improved. The following two methods can be used to set a threshold for source reliability. (a) The average of SR for all sources is determined and then can be used as a threshold to determine only those sources which have above average source reliability. (b) On the other hand, the number of opinions a source has about a specific or all features of a product can also be used as a threshold. For example, a threshold of 20 opinions can be set to determine only those sources which have at least 20 opinions about a feature.

Similarly, for computing "features based product reputation" value, source reliability $\mathrm{SR}(s, F, g)$ of source s for features set $F = \{f_1, f_2, f_3, \ldots, f_q\}$, about product g is computed using Eq. (4.3).

$$\mathrm{SR}(s, F, g) = \frac{\prod_{k=1}^{q}(\mu(s, f_k, g) + 1)}{\sum_{i=1}^{d}(\prod_{k=1}^{q}(\mu(s_i, f_k, g) + 1))}, \tag{4.3}$$

where μ is the number of opinions of feature f posted on source s about product g and q is the total number of features considered for determining the "features based product reputation" value.

In a more personalized setting, when a user want to compute reputation based on few preferred features PF, we proposed Eq. (4.4) to compute $\mathrm{SR}(s, PF, g)$ of source s for preferred feature set $\mathrm{PF} = \{f_1, f_2, f_3\}$ of product g. In this case, the sources that do not have ratings about any preferred feature is given weight 0 and the reputation will be based on sources that have ratings about all preferred features.

$$\mathrm{FC}(s, \mathrm{PF}, g) = \frac{\mu(s, f_1, g) \times \mu(s, f_2, g) \times \mu(s, f_3, g)}{\sum_{i=1}^{d}(\mu(s_i, f_1, g) \times \mu(s_i, f_2, g) \times \mu(s_i, f_3, g))}. \tag{4.4}$$

4.2. *Reviewer expertise*

Reviewer expertise is the degree to which a reviewer is capable to provide valuable information. It is very important to consider the reviewer expertise because higher the expertise of a user, higher the valuableness of the evaluation he would provide.[21] An expert user knows the product very well, used different aspects of the product and aware of functionalities and technicalities of the product, hence expert user is more reliable evaluator. Some users even pay to avail the services of experts to get their point of view about product. Different websites offer this service in order to help the customers to make purchase decision.

We propose two different methods to determine the expertise of a reviewer. In first method, we determine the expertise of reviewer by using the number of helpfulness votes that a reviewer received for his review. It is observed that the review which provides valuable information to customers gets more positive helpfulness votes and is written by an expert user.

We proposed Eq. (4.5) to compute expertise $\mathrm{RE}(b, r, g)$ of reviewer b for their review r about product g, using the number of positive helpfulness votes α and negative helpfulness votes β issued by other users which are extracted from e-commerce and review sites.

$$\mathrm{RE}(b, r, g) = \frac{\alpha(r)}{\alpha(r) + \beta(r)} \div \sum_{i=1}^{n} \left(\frac{\alpha_i}{\alpha_i + \beta_i} \right). \tag{4.5}$$

The issue with this method is that sometimes very few reviews which are posted earlier get review helpfulness votes and the rest have no helpfulness votes. Actually, it takes time to get helpfulness votes. A recent review even posted by an expert user may not have helpfulness votes. In order to address this issue, we also proposed another method to compute reviewer expertise. We observed that expert users know well about different product features, product functionalities and technicalities, hence discuss most of the features in details in their reviews. On the other hand, the reviewers who list down several product features in one or two sentences are not expert users. Therefore, we introduced another measure "the number of features discussed in detail" to rank the expertise of a reviewer. For this purpose, the product features identified in Sec. 3.2 are used and the numbers of sentences where these features are discussed are determined in a review using the Apache OpenNLP[h] sentence detector. A sentence is count once when more than one feature are discussed in it. Higher the number of features discussed in separate sentences by a reviewer, higher will be the expertise of that reviewer. Equation (4.6) is proposed which uses the number of sentences λ where unique features of the product is discussed, where q is the total number of features.

$$\mathrm{RE}(b, r, g) = \frac{\lambda(r)}{q} \div \sum_{i=1}^{n} \left(\frac{\lambda_i}{q} \right). \tag{4.6}$$

The $\mathrm{RE}(b, r, g)$ can be used in two ways. First, it can be used to give more weight to expert reviewers and less weight to nonexpert reviewers as computed by Eq. (4.5) or (4.6). Second, this value can also be used to form a group of qualified expert reviewers based on a threshold which determines that whether a reviewer meet a certain level of expertise or not. The threshold can be set using any one of the following two methods based on the user preferences. (a) The average reviewer expertise ARE is computed by adding the review expertise RE of all reviews divided by the total number of reviews n. The ARE is then used as a threshold to determine only those reviewers which have at least above average expertise level. While computing the reputation value, only the ratings posted by the reviewers who meet above average level of expertise will be considered. (b) The threshold can also be set on the bases of number of features discussed by a reviewer. For example, if there are 10 features of a product then a threshold of five features can be used in order to consider only those reviewers who discussed at least five features of the product. The λ used in

[h] http://opennlp.apache.org/.

Eq. (4.6) is used to implement this threshold. The reviewers who have λ equal or greater than five for his review will meet the required expertise level because they have discussed at least five features.

4.3. *Rating trustworthiness*

Rating trustworthiness measures the degree to which a reviewer provides honest rating which reflects its own feeling and opinion. In other words, we can say that rating trustworthiness determines that whether the review is fake or genuine. Several terms for ratings trustworthiness are used in the literature such as spam detection, fake reviews detection or opinions fraud detection.[16–18] In fact, on e-commerce and product reviews sites, some reviewers post biased or spam reviews in order to manipulate reputation. Both types of biased behavior that is under rating and over rating exist in product reviews. Unfortunately, most of product reputation systems do not consider rating trustworthiness while determining reputation and hence not able to deal with biased behaviors.

In order to incorporate rating trustworthiness in our model, we implemented the method proposed by Jindal and Liu[16] with little variations. Three types of spam reviews defined in Ref. 16 which are found on e-commence and review sites, i.e., untruthful reviews (type 1), reviews on brand only (type 2) and nonreviews (type 3). Untruthful reviews are those reviews which are deliberately posted either to promote a product or to damage the reputation of a product. Reviews on brand are those that comments on brand, manufacturer or seller instead of expressing opinions about product. On the other hand, nonreviews contain irrelevant text such as advertisement, questions and answers or random text. The two-gram based content comparison of reviews is performed in order to determine the duplicate reviews in the same way as in Ref. 16. A two-gram is actually a fixed window of two words which appeared successively. The similarity between two reviews is the ratio of the common two-grams in both reviews and the union of two-grams of both reviews, which is usually known as Jaccard distance.[22] Let the two-grams of two reviews r_1 and r_2 are $M_2(r_1)$ and $M_2(r_2)$ then the Jaccard distance $\text{sim}(r_1, r_2)$ can be determined using Eq. (4.7). Reviews that have similarity at least 80% are considered duplicate.

$$\text{sim}(r_1, r_2) = \frac{|M_2(r_1) \cap M_2(r_2)|}{|M_2(r_1) \cup M_2(r_2)|}. \tag{4.7}$$

On the other hand, in order to detect the spam reviews of type 2, type 3 and type 1 which are not duplicated, a model is built based on the features which provide the likelihood that a review is to be spam. Three main categories of features are defined (i.e., review centric features, reviewer centric features and product centric features) in Ref. 16, most of which are manually labeled. However, we are using those subfeatures (i.e., reviewer id, reviewer name, textual rating, numeric rating, review helpfulness, ASV, brand name, product features, etc.) of these three main categories,

which can be determined automatically from the information extracted from review sites. The model is built with logistic regression to produce a probability estimation which reflects the likelihood that the review is spam. The *statistical package* R[i] is used for logistic regression. The result of logistic regression (which represents the probability that a review is spam) is weighted down in order to determine rating trustworthiness RT. On the other hand, the rating trustworthiness for each duplicate review is assign value zero because intent in such kind of reviews is to manipulate reputation value.

4.4. *Ageing factor*

The opinions of people about a product may change with the passage of time.[20] This change may cause by several reasons such as rapid change in technology, change in customer needs, market trend and change in quality of the product. A reputation model must be able to reflect these variations in opinions in order to determine the most recent reputation. Indeed, the recent ratings are more important than old ratings and hence need to be given more weight over older one. This can be achieved by introducing an ageing factor called linear decay describe in our previous study.[20] The linear decay is selected because it satisfies all constraints related to ageing factor (as defined in the literature review) and hence more suitable in product reputation context.

Let t_i is the time (Review Date) on which review r_i is posted, where $t_i \in T = \{t_1, t_2, t_3, \ldots, t_n\}$ and $r_i \in R = \{r_1, r_2, r_3, \ldots, r_n\}$. Then the weight w_i associated with rating r_i is computed using Eq. (4.8), where $w_i \in W = \{w_1, w_2, w_3, \ldots, w_n\}$. The w_1 is the weight assigned to the most recent rating which is 1 and w_n is the weight assigned to oldest rating which is equal to 0, l is the number of unique times on which ratings are posted and n is the total number of ratings about product g or about a feature f. Equation (4.9) is used to obtain normalized weight AF_i which is also called the ageing factor.

$$w_i = w_1 + (i-1)\left(\frac{(w_n - w_1)}{(l-1)}\right), \tag{4.8}$$

$$\mathrm{AF}_i = \frac{w_i}{\sum_{i=1}^{n} w_i}. \tag{4.9}$$

5. Product and Feature Reputation

This section explains the proposed aggregation methods to compute different reputation values. Five reputation values: feature reputation, feature-based product reputation, aggregated general opinions (AGO), product reputation and product reputation based on review titles are computed.

[i] https://www.r-project.org/.

5.1. *Feature reputation*

The main focus of the proposed model is to evaluate product based on features. One of the most important reputation values is feature reputation which is very useful for both customer and manufacturer to make decisions. Sometimes, the customers want to make decision on the bases of certain features of the product. For example, if a customer want to buy a phone with high screen resolution, then he need to check the reputation of feature screen of different mobiles to make a purchase decision. Similarly, the manufacturers are also interested to know the customers' opinions about different features of the product in order to identify the strengths and weaknesses of the product.

In order to compute feature reputation, we use the result of sentiment analysis which is in the form of positive and negative opinions while considering the corresponding rating reputation. We proposed Eq. (5.1) to compute a feature reputation.

$$R(f) = \frac{\left(\sum_{i=1}^{\mu} \mathrm{RR}_i \times \mathrm{PO}_i\right)}{\left(\sum_{i=1}^{\mu} \mathrm{RR}_i \times \mathrm{PO}_i + \sum_{i=1}^{\mu} \mathrm{RR}_i \times \mathrm{NO}_i\right)} \times 100, \tag{5.1}$$

where $R(f)$ is the reputation of feature f, RR_i is the rating reputation of opinion i computed using Eq. (4.1), PO and NO are the positive and negative opinions respectively about feature f and μ is the total number of opinions about the same feature. When a user want to compute $R(f)$ by only considering one parameter (i.e., SR or RE or AF) then the RR_i is replaced with that parameter.

The feature reputation value computed using Eq. (5.1) is within the range of 1–100. This value contains more information and hence more useful for the manufactures in order to make decisions. However, this value is difficult to understand for customers because the online ratings systems use five stars scale and hence the customers are very familiar with this rating mode. In addition, the authors in Ref. 23 compared different ratings modes and suggested that rating mode with five or seven scales is more appropriate for human to understand and they are also easy to map into linguistic variables. On the other hand, in order to aggregate different reputation values, we need to bring all reputation values into one scale. As we know that ASV[20] is in five star scale, however the feature reputation is in 1–100 scale. Therefore, Eq. (5.2) is proposed to convert feature reputation value $R(f)$ in 1–100 scale into five stars scale $R(f, 5)$ and vice versa.

$$y = \left\lceil \frac{x}{20} \right\rceil, \tag{5.2}$$

where x represents the reputation value in 1–100 scale and y represents the corresponding five stars scale value. The floor function $\lfloor \frac{x}{20} \rfloor$ introduced in Ref. 24 is used to round the number $\frac{x}{20}$ to the next larger integer.

5.2. *Features based product reputation*

The "features based product reputation" value depends on the reputation of different features of the product. This value is very useful when the user want to

compare different products on the bases of features. Sometimes, the users are looking for a product with best features. In such case, the users can use this reputation value to compare product based on some features in order to make a purchase decision.

Instead of using the existing aggregation methods which have several issues, we adapted the weighted median so that it can be used for computing *feature based product reputation*. Weighted median offers several benefits over existing aggregation methods such as it inherent the robustness and strategy proofness characteristics of classical median and it also does not need tie breaking rules. In addition, the weighted media also allows us to aggregate information while considering the importance of each observation. The "features based product reputation" is computed by aggregating five stars scale reputation values of all features of a product while considering the importance of each feature using weighted median. First, the importance of each feature is computed. We know that all features of a product are not equally important because some features are more important than others. Therefore, the importance of different features is considered while computing "feature-based product reputation". The importance of a feature depends on its popularity in user generated text.[8] The features that are more frequent in user generated text are considered to be more important. We also used the feature frequency to compute feature importance; however a normalized value is computed. We proposed Eq. (5.3) to compute importance of each feature which we also called feature weight.

$$\mathrm{FW}(f) = \frac{\mu(f)}{\mu(f, \max)} \div \left(\sum_{i=1}^{q} \frac{\mu(f_i)}{\mu(f, \max)} \right), \tag{5.3}$$

where $\mathrm{FW}(f)$ is the feature weight f, $\mu(f)$ is the frequency of feature f in user generated text, $\mu(f, \max)$ is the frequency of most frequent feature of the same product and q is the total number of features.

After computing feature importance, now we can compute "features-based product reputation" $R(P, F)$, by aggregating feature reputation values $\{R(f_1, 5), R(f_2, 5), R(f_3, 5), \ldots, R(f_q, 5)\}$ of all features of a product while considering the feature importance $\{\mathrm{FW}(f_1), \mathrm{FW}(f_2), \mathrm{FW}(f_3), \ldots, \mathrm{FW}(f_q)\}$ of each feature using weighted median. The following steps are followed in order to obtain the weighted median.

(i) Sort the $R(f, 5)$ of all product features in descending order,
Let $R(f_1, 5) \geq R(f_2, 5) \geq R(f_3, 5) \cdots \geq R(f_q, 5)$
(ii) Let the corresponding normalized feature weights are:
$\mathrm{FW}(f_1), \mathrm{FW}(f_2), \mathrm{FW}(f_3) \cdots \mathrm{FW}(f_q)$
(iii) Find FW $(f_{k\Psi})$ such that Eqs. (5.4) and (5.5) hold.

$$\mathrm{FW}(f_{k\Psi}) + \sum_{k=1}^{k\Psi} \mathrm{FW}(f_k) \geq \sum_{k=k\Psi+1}^{q} \mathrm{FW}(f_k), \tag{5.4}$$

$$\sum_{k=1}^{k\Psi-1} \mathrm{FW}(f_k) \leq \sum_{k=k\Psi}^{q} \mathrm{FW}(f_k) + \mathrm{FW}(f_{k\Psi}). \tag{5.5}$$

(iv) Then the $R(f_{k\Psi}, 5)$ value associated with FW $(f_{k\Psi})$ is the weighted median which is the $R(P, F)$.

In more personalized setting, the users may be interested to compute the product reputation $R(P, \mathrm{PF})$ based on few preferred features PF. This value is very important to compare different products in order to select the best product on the bases of preferred features. For example, when a user wants to compare different mobile phones on the bases of features such as battery, screen and processing speed then the user can compute $R(P, \mathrm{PF})$ only based on these features. The same aggregation method as proposed for "features-based product reputation" is used, however the customers are provided with three choices to assign feature weight to all three features based on their need. (a) When all the features are equally important for user, then it is appropriate to assign the same weight to each feature (e.g., FW $(f_i) = 1$). (b) When the user has its own preferences within the preferred features, then he can assign weights manually to each feature according to their preference. (c) On the other hand, when the user want to decide on the bases of feature popularity then the proposed method can be used to compute FW (f) for each feature.

5.3. *Aggregating general opinions*

A large ratio of sentiments in textual ratings is generally about product without targeting some specific features of the product.[8] For example, the textual comments such as "the mobile is nice" and "it is also reliable" the reviewer is expressing their opinions generally about mobile without targeting any specific feature such as screen, battery or camera, etc. These opinions also need to be considered while determining the product reputation. The general opinion about product is a general subjective measure which is usually given on the bases of average performance of the product. However, sometimes few features are more important for users and the opinions about those features dominate the overall opinion about product.

In order to aggregate general opinions, the results of sentiment analysis (i.e., the positive and negative opinions generally about product) are aggregated while considering rating reputation. The same aggregation method is used as for feature reputation, however here the general sentiment orientations about product are considered. Equation (5.6) is proposed to compute AGO value.

$$\mathrm{AGO} = \frac{(\sum_{i=1}^{e} \mathrm{RR}_i \times \mathrm{GPO}_i)}{(\sum_{i=1}^{e} \mathrm{RR}_i \times \mathrm{GPO}_i + \sum_{i=1}^{e} \mathrm{RR}_i \times \mathrm{GNO}_i)} \times 100, \tag{5.6}$$

where GPO is positive opinions generally about product, GNO is the negative opinion generally about product and e is the total number of general opinions about product.

5.4. *Product reputation*

This reputation value represents the overall product reputation and based on both numeric and textual ratings. Considering both numeric and textual ratings increases the computation feasibility because sometimes either numeric or textual ratings alone is not enough to compute a reputation value. This also increases the reliability because the reputation is based on different types of ratings. Three different reputation values: ASv,[20] feature-based reputation $R(P, F)$ and AGO are aggregated to compute overall product reputation value. The ASv is obtained by aggregating the numeric ratings using weighted median as described in our previous study.[20] The $R(P, F)$ and AGO are computed using the method proposed in Secs. 5.2 and 5.3, respectively. The AGO is converted into five stars scale AGO(5) using Eq. (5.2). Simple weighted mean is used to compute product reputation. As we have produced a layer of robust aggregation methods below, therefore using weighted mean above does not compromise the robustness to false ratings. Equation (5.7) is used to compute overall product reputation $R(P)$.

$$R(P) = \frac{w_{\text{ASv}} \times \text{ASv} + w_{R(P,F)} \times R(P, F) + w_{\text{AGO}(5)} \times \text{AGO}(5)}{\sum_{i=1}^{c} w_i}, \tag{5.7}$$

where w_{ASv}, $w_{R(P,F)}$ and $w_{\text{AGO}(5)}$ are the weights associated with three reputation values (i.e., ASv, $R(P, F)$ and AGO(5)). The values of these weights need to be assigned according to the following issues related to both numeric and textual ratings.

(i) Usually, the numeric ratings are more biased because the malicious users post unfair ratings to obtain their preferred ASv. In such case, we need to give more weight to the values obtained from textual ratings.
(ii) On the other hand, the sentiment analysis method used to analyze textual ratings may not give satisfactory results for a specific product. In such situation, the value obtained from numeric ratings (i.e., ASv) need to be given more weight.
(iii) Sometimes, products have very few reviews and the opinions about features is not enough to compute $R(P, F)$, in this case we need to compute reputation only based on numeric ratings which can be achieved by assigning zero weight to the reputation values based on textual ratings (i.e., $w_{R(P,F)} = 0$, $w_{\text{AGO}(5)} = 0$ and $w_{\text{ASv}} = 1$)).
(iv) We consider the optimal case when the results of sentiment analysis is satisfactory, the numeric ratings are rarely biased and we have enough number of opinions for computing $R(P, F)$. Therefore, the product reputation value that we have computed is equally based on numeric and textual ratings (i.e., $w_{R(P,F)} = 0.25$, $w_{\text{AGO}(5)} = 0.25$ and $w_{\text{ASv}} = 0.5$).

5.5. *Product reputation based on review titles*

The review titles where the customers either summarize the product or provide some important aspect of the product in few words or in a sentence are also very handy in

context of product reputation. Some sort of textual analysis can be used to summarize the review titles in order to produce a summary about the most important aspects of the product. However, in our model we only compute a reputation value based on the results of sentiment analysis of reviews titles while considering rating reputation. Equation (5.8) is proposed in order to compute product reputation based on review titles $R(P, \mathrm{rt})$.

$$R(P, rt) = \left\lceil \left(\frac{\sum_{i=1}^{p} \mathrm{RR}_i \times \mathrm{Prt}_i}{\sum_{i=1}^{p} \mathrm{RR}_i \times \mathrm{Pr}t_i + \sum_{i=1}^{p} \mathrm{RR}_i \times \mathrm{Nrt}_i} \times 100)/20 \right) \right\rceil, \tag{5.8}$$

where Prt and Nrt are the positive and negative review titles respectively, p is the total number of review titles where opinions are expressed and the ceiling function $\lceil x \rceil$ is used to round the number to the next larger integer.

6. Experiments and Results

In order to validate the results and evaluate the performance of our reputation model three evaluation criteria, i.e., robustness, sensitivity and estimation accuracy are used. The results of proposed model based on these evaluation criteria are compared with existing product reputation models and different aggregation methods.

6.1. *Sensitivity*

Sensitivity measures the degree to which a reputation model or aggregation method reflects the recent ratings. For example, if a product or a feature has good reputation. However, due to some reasons, the quality of the product/feature become worst and the reviewers started to rate it with lowest ratings. In such case, the sensitivity will measures the ability of an aggregation method that how fast it reflects this change in order to compute the recent reputation. There is a trade-off between sensitivity and robustness. Therefore, an aggregation method must produce a balance between sensitivity and robustness. A too much sensitive aggregation method can be easily compromised by malicious users whereas a too much robust aggregation method will not be able to reflect the recent opinions quickly. A more suitable situation can be when an aggregation method is robust to false and biased ratings but fairly sensitive to genuine ratings.

In order to measure the sensitivity of aggregation methods, reviews of 25 products are extracted from reviews sites such as Ebay, Cnet and Amazon using the review extraction method. Product features are identified using the proposed method in Ref. 3. The opinions about different features are determined (i.e., whether the opinions are positive or negative) using the sentiment analysis method discussed in Sec. 3.2. The feature reputation value $R(f, 5)$ for all features in five stars scale are computed using our proposed method. A combination of quota sampling[25] and systematic sampling[26] methods is used to select a sample of features for the experiment. The quota sampling is used to segment all features first into mutually exclusive

subgroups based on good, average and bad features reputation and then only features with good and bad reputation are select. The feature which has $R(f, 5)$ value at least 4 (i.e., 4 or 5) is considered as feature with good reputation, whereas features with bad reputation are those which has $R(f, 5)$ less than or equal to 2 (i.e., 2 or 1). The features with good and bad reputation are further sampled separately using systematic sampling method, where features with different number of opinions after a regular interval are selected. For example, the features with bad reputation are first arranged by the total number of opinions and then every 20th feature is selected in ordered list (i.e., 20, 40, 60, ..., 200) as shown in Table 3. We used opinion and rating interchangeably because opinion is actually textual rating. Two methods are used to determine the sensitivity. For features with good reputation (i.e., f-1, f-2, f-3, ..., f-10), we assumed that the opinions of customers about that feature changed due to some reasons and hence they started to express negative opinions. The sensitivity of aggregation methods is measured by counting the number of negative opinions needed to reflect the recent opinions fully; i.e., to bring feature reputation from good (5 or 4 star) to worst (1 star). On the other hand, for the features with bad reputation (i.e., f-11, f-12, f-13, ..., f-20) it is assumed that the opinion changed and the reviewers started to express positive opinions. The sensitivity is measured by counting the number of positive opinions needed to bring the feature reputation from 1/2 star to 5 star. In order to present the results in more meaningful way, we further determined the percent of the existing ratings (which are posted before the change in opinions of customers) needed to fully reflect the recent opinions. For example, if a feature has μ number of opinions before the change in customers' opinions and v are the number of opinions needed after the change in customers' opinions to fully reflect the recent opinion, then the percentage of existing ratings μ is equal to $v/\mu * 100$. In order to compare the results of proposed aggregation method with existing methods, the sensitivity is measured in five star scale for all aggregation methods. We computed the results with some changes for the aggregation method RPOAO[8] which used the RPOAO to compute feature reputation. First, in this method the frequent negative opinions are given more weight, however due to comparison with other aggregation methods we used the same weight for both frequent and infrequent opinions. Second, a manual hierarchy of features and subfeatures are built and positive opinions of sub features are also included while computing reputation of a feature in this method. However, we computed the result of each feature separately without building a hierarchy manually, which allows us to compare with other aggregation methods more precisely.

The sensitivity results of the proposed aggregation method for feature reputation are compared with existing methods RPOAO[8] and DPNOAO[9] in Tables 2 and 3. The DPNOAO used the difference of positive and negative over all opinions in order to compute feature reputation. Table 2 shows the sensitivity results for features with good reputation whereas Table 3 shows the results for feature with bad reputation. The results show that the proposed method reflects the recent opinions more quickly than other methods because on average 64.6% and 71.1% of the existing ratings are

Table 2. Sensitivity of aggregation methods for features with good reputation.

Feature	f-1	f-2	f-3	f-4	f-5	f-6	f-7	f-8	f-9	f-10	Average
Total no. of opinions	10	30	50	70	90	110	130	150	170	190	
RPOAO[8]	240	156.7	142	154.3	134.4	145.5	211.5	214	208.8	202.6	181
DPNOAO[9]	260	236.7	210	222.9	205.6	218.2	230.8	233.3	224.1	226.3	226.8
DPNOAO+AF	80	70	72	78.6	73.3	75.5	80.8	84.7	82.4	81.6	77.9
Proposed method	70	50	50	55.7	51.1	59.1	76.9	78	76.5	78.9	64.6

Table 3. Sensitivity of aggregation methods for features with bad reputation.

Feature	f-11	f-12	f-13	f-14	f-15	f-16	f-17	f-18	f-19	f-20	Average
Total no. of opinions	20	40	60	80	100	120	140	160	180	200	
RPOAO[8]	210	200	195	138.8	151	240	203.6	218.8	242.8	210	201
DPNOAO[9]	310	287.5	285	220	225	258.3	218.6	234.4	261.7	225.5	252.6
DPNOAO+AF	95	95	96.7	70	78	83.3	77.1	78.1	94.4	80	84.8
Proposed method	75	72.5	73.3	46.3	56	76.7	71.4	75	88.9	76	71.1

needed. Conversely, the other aggregation methods[8,9] needed very large percentage of opinions (i.e., 181 and 201 and 226.8 and 252.6, respectively) to bring the feature reputation from good to worst or vice versa. In both these methods, for single features, at least 134% of opinions are needed to reflect the recent opinions which is too much. We also used the ageing factor AF computed in Sec. 4.4 with DPNOAO (DPNOAO+AF), however this combination also needed relative larger percentage of opinions than the proposed method.

6.2. *Robustness*

An aggregation method is said to be robust if the false ratings may not easily affect the aggregated result. Higher the number of false and biased ratings needed to change the aggregation result, more robust the aggregation method will be. In order to perform the experiment for determining the robustness of aggregation methods, reviews of 20 products are extracted from reviews sites. Product features are identified using the proposed method in Ref. 3. The opinions about different features are determined using the sentiment analysis method discussed in Sec. 3.2. Ten features (f-1, f-2, f-3, ..., f-10) are selected for the experiment which have different number of ratings (i.e., total number of opinions as shown in Table 4) and feature reputation value. Usually the robustness is measured by inserting false and biased ratings until the result of an aggregation method change. However, in order to know that how a reputation model behaves in different scenarios and to different types of rating we insert four types of ratings samples to measure the robustness. The ratings samples include: (1) positive false or biased ratings (2) negative false or biased ratings (3)

Table 4. Robustness against negative biased ratings.

Feature	f-1	f-2	f-3	f-4	f-5	f-6	f-7	f-8	f-9	f-10	Average
Total no. of opinions	10	20	30	40	50	60	70	80	90	100	
RPOAO[8]	4	2	12	23	6	26	36	12	40	6	16.7
DPNOAO[9]	2	2	13	24	7	33	35	13	45	8	18.2
PM+RT	7	15	34	45	35	25	45	33	42	37	31.8
PM+RT+SR	19	35	45	50	37	35	47	38	58	45	40.9
PM+RT+SR+AF	8	15	22	25	15	22	30	31	30	35	23.3

negative ratings uniformly distributed which contains all types of ratings (i.e., false, biased, duplicate, genuine, etc.) and (4) positive genuine ratings. In all these scenarios, the robustness is measured by counting the number of ratings needed to change the feature reputation by one star. For example, if a feature has reputation value of three star, then how many positive ratings will be needed for an aggregation method to change the value to four star. The first two ratings samples are used in order to measure the robustness of aggregation methods to false and biased ratings. The third ratings sample is used in order to determine that how the aggregation methods react if the ratings are uniformly distributed. The last ratings sample is used in order to determine that how the aggregation methods behave to genuine ratings. In order to compare the results of proposed aggregation method with existing methods, the robustness is measured in five star scale for all aggregation methods.

Table 4 shows the number of negative false and biased ratings to change the result of aggregation methods for feature reputation. The results of the proposed method are compared with existing methods for feature reputation, i.e., RPOAO and DPNOAO. Three set of results are obtained using the proposed method PM by considering one or more rating parameters as shown in Table 4, i.e., (1) PM+RT, (2) PM+RT+SR and (3) PM+RT+SR+AF. The PM+RT represents the results when only rating trustworthiness RT is considered with the proposed aggregation method, the PM+RT+SR represents the results when both rating trustworthiness and source reliability SR are considered, while PM+RT+SR+AF are the results when aging factor AF is also used. The results show that the existing methods RPOAO and DPNOAO are less robust to positive false and biased ratings because only 16.7 and 18.2 negative ratings on average are needed to change the reputation value. The results also show that the proposed method is more robust to false and biased ratings. When only rating trustworthiness is considered (i.e., PM+RT), then on average 31.8 negative false and biased ratings are needed. The robustness of the proposed method increased to 40.9 ratings when the source reliability is also considered because not only false and biased ratings are discounted but also the ratings from less opinionated sources. However, when aging factor is also considered with other parameters (i.e., PM+RT+SR+AF), then the robustness of the proposed method decreased to 23.3 because the recent ratings are favored, but it is still more robust than the existing methods. On the other hand, Table 5 shows the number of positive false and biased

Table 5. Robustness against positive biased ratings.

Feature	f-1	f-2	f-3	f-4	f-5	f-6	f-7	f-8	f-9	f-10	Average
Total no. of opinions	10	20	30	40	50	60	70	80	90	100	
RPOAO[8]	6	9	22	10	17	18	21	19	27	7	15.6
DPNOAO[9]	6	12	30	11	23	17	20	17	25	4	16.5
PM+RT	17	30	11	18	20	27	35	27	55	40	28
PM+RT+SR	27	35	22	30	35	30	36	40	47	50	35.2
PM+RT+SR+AF	6	15	8	16	17	22	33	24	28	32	20.1

ratings to change the result of aggregation methods for feature reputation. Same patterns of results are obtained for the existing methods as well as for the proposed method with different parameters. In Conclusion, we can say that Tables 4 and 5 show that the proposed method is more robust to both positive and negative biased ratings.

Table 6 shows the robustness of aggregation methods against negative ratings which are uniformly distributed. The results of the existing methods RPOAO and DPNOAO remained the same as in Table 4. This shows that existing methods behaved in the same way whether all the ratings are biased or some of them are biased. Conversely, the robustness of the proposed method decreased compared to the previous experiment (i.e., Table 4) because some ratings are genuine which are favored. This shows that the proposed method reacted differently to biased and uniformly distributed ratings. It also depends on the distribution of ratings in the rating sample. If the rating sample contains more biased ratings, then the proposed method is more robust otherwise less robust if it contains more genuine ratings.

Table 7 shows the robustness of aggregation method against positive genuine ratings. The results of the existing methods remained the same as in Table 5 because the existing methods are not able to differentiate between biased and genuine ratings and hence are not able to reflect genuine ratings quickly. On the contrary, the proposed method is sensitive to genuine ratings because on average only 8.2 positive genuine ratings are needed to change the reputation value. At the same time, it also not too much sensitive to genuine ratings and hence the malicious reviewers who can bypass spam review detection cannot easily change the reputation value. On the

Table 6. Robustness against negative, uniformly distributed ratings.

Feature	f-1	f-2	f-3	f-4	f-5	f-6	f-7	f-8	f-9	f-10	Average
Total no. of opinions	10	20	30	40	50	60	70	80	90	100	
RPOAO[8]	4	2	12	23	6	26	36	12	40	6	16.7
DPNOAO[9]	2	2	13	24	7	33	35	13	45	8	18.2
PM+RT	5	18	22	35	12	45	50	21	25	35	26.8
PM+RT+SR	6	20	32	50	20	50	33	30	28	39	30.8
PM+RT+SR+AF	4	5	17	20	10	31	21	20	18	25	17.1

Table 7. Robustness against positive genuien ratings.

Feature	*f*-1	*f*-2	*f*-3	*f*-4	*f*-5	*f*-6	*f*-7	*f*-8	*f*-9	*f*-10	Average
Total no. of opinions	10	20	30	40	50	60	70	80	90	100	
RPOAO[8]	6	9	22	10	17	18	21	19	27	7	15.6
DPNOAO[9]	6	12	30	11	23	17	20	17	25	4	16.5
Proposed method	4	4	10	6	10	8	10	7	12	8	8.2

bases of all these results about robustness, we can conclude, that the proposed method is more robust to false and biased ratings. In addition, it also behaves differently to different types of ratings and hence more adoptable to obtain desirable results according to the scenario.

6.3. *Estimation accuracy*

The estimation accuracy measures the degree to which a reputation model or aggregation method accurately estimates a true reputation value. The AE is used to compute the accuracy, which consists of computing the absolute value of the difference between the actual and the predicted reputation values.[27] The predicted reputation value is determined using our proposed model. However, the actual reputation value is computed using three field experts which are provided with ratings and the associated ratings parameters (i.e., rating trustworthiness, source reliability and reviewer expertise) and rating reputation. The intraclass correlation coefficient,[28] abbreviated as ICC is used to compute the agreement level between these experts. Type ICC(3,K) of ICC was adopted, where each subject (i.e., feature reputation value or "feature-based product reputation" value) is assessed by three experts and the agreement is calculated by taking an average of k experts measurements. The statistic application called SPSS[29] is used for computing ICC. The experts agreement for determining feature reputation for features shown in Tables 8 and 9 are 87% and 82%, respectively. Similarly, the agreement between experts while determining feature-based product reputation value for products shown in Table 10 is 78%. Based on ICC interpretation, we can say that there is almost perfect agreement (i.e., $> 80\%$) between experts while determining feature reputation value and substantial agreement (i.e., between 60–80) while calculating feature base

Table 8. Estimation accuracy of feature reputation 1.

Feature	Absolute error										MAE
	f-1	*f*-2	*f*-3	*f*-4	*f*-5	*f*-6	*f*-7	*f*-8	*f*-9	*f*-10	
Total no. of opinions	25	35	45	50	60	70	85	95	105	120	—
RPOAO[8]	0.06	0.09	0.05	0.14	0.06	0.11	0.13	0.04	0.12	0.08	0.088
DPNOAO[9]	0.06	0.08	0.05	0.14	0.06	0.11	0.13	0.03	0.12	0.08	0.086
Proposed method	0.02	0.04	0.06	0.04	0.02	0.1	0.04	0.08	0.07	0.05	0.052

Table 9. Estimation accuracy of feature reputation 2.

Feature	Absolute error										MAE
	f-11	f-12	f-13	f-14	f-15	f-16	f-17	f-18	f-19	f-20	
Total no. of opinions	45	50	60	70	85	95	105	120	130	150	—
RPOAO[8]	0.22	0.16	0.24	0.15	0.11	0.12	0.17	0.24	0.22	0.32	0.195
DPNOAO[9]	0.23	0.16	0.24	0.15	0.11	0.11	0.17	0.24	0.21	0.31	0.193
Proposed Method	0.01	0.04	0.03	0.05	0.03	0.04	0.06	0.14	0.05	0.04	0.049

product reputation value. The average accuracy is computed by mean absolute error (MAE),[30] which determines the average of all AEs. Lower the value AE and/or MAE for an aggregation method, higher the accuracy. AE and MAE are computed for different aggregation methods using Eqs. (6.1) and (6.2), respectively. Where R_a is the actual reputation value, R_p is the predicted reputation value and q is the total number of features for which AE is computed.

$$\text{AE} = |R_a - R_p|/100, \tag{6.1}$$

$$\text{MAE} = \frac{\sum_{i=1}^{q} \text{AE}_i}{q}. \tag{6.2}$$

6.3.1. *Estimation accuracy of aggregation method for feature reputation*

In order to measure the accuracy of proposed aggregation method for feature reputation, reviews of 15 products are extracted from different sources such as Amazon, Cnet and Ebay. Ten features (f-1, f-2, f-3, ..., f-10) which have different number of opinions are selected and sentiment analysis is used to determine that whether the opinions about these features are positive or negative. Some of the web sources have fewer numbers of opinions and some opinions belong to reviews which are not trustworthy. Equations (6.1) and (6.2) are used to compute AE and MAE. The results of all aggregation methods are obtained in 1–100 scale. Equation (6.3) is used to convert the reputation value computed in 1–1 scale into 1–100 scale, where z is the reputation value in 1 to −1 scale and x is the corresponding value in 1–100 scale.

$$\begin{cases} \text{if } z \geq 0, & x = ((z * 100) + 100)/2, \\ \text{if } z < 0, & x = ((z * 100) + 101)/2. \end{cases} \tag{6.3}$$

Table 8 shows the results of estimation accuracy of the aggregation methods for feature reputation. The results show that average accuracy of the proposed methods is better than the existing methods. The MAE of the proposed method is 0.056, however the MAE of the RPOAO and DPNOAO is 0.096 and 0.094, respectively. The results also show that the proposed method offers the best accuracy AE in most of the features (i.e., in eight out of ten). The proposed method offered the best

accuracy because it discounts both false ratings and ratings from less opinionated sources as well as favors recent ratings.

We performed another set of experiments using the same experimental setting with a little variation in order to measure the accuracy in situations when there is a shift in opinions of customers. Ten features (f-11, f-12, f-13, ..., f-20) in which five already maintained good reputation and the same number of features maintained bad reputation. We considered that due to some reasons, the opinions of customers changed and most of the customers started to express negative opinions about features with good reputation and positive opinions about features with bad reputation. Table 9 shows the results of this experiment. The results show that the average accuracy of both RPOAO and DPNOAO become worst when there is a shift in customers' opinions. The MAE of RPOAO and DPNOAO are increased to 0.195 and 0.193 respectively which is almost double of the results of previous experiment as shown in Table 8. Our proposed aggregation method still maintained a good average accuracy as the MAE is 0.049. This shows that the proposed aggregation method not only achieved good accuracy in ordinary situation but also when there is a shift in customers' opinions. The reason behind achieving good accuracy in such situations is that, the proposed method favors recent ratings through ageing factor. This also shows that if an aggregation method does not favor recent ratings then the reputation value can be misled in such situations.

6.3.2. *Estimation accuracy of aggregation method for features based product reputation*

In order to measure the accuracy of aggregation method for "features based product reputation", we have selected features of four products (P-1, P-2, P-3 and P-4) after the same data processing and same method for computing feature reputation as in Sec. 6.3.1.

The results of proposed method is compared in Table 10 with existing method W. Mean+RPOAO[8] and other combinations of existing methods, i.e., Mean+RPOAO, Mean+DPNOAP and W.Mean+DPNOAP. In all these methods, actually a combination of two aggregation methods are used, the first aggregation method (which is either mean or weighted mean) is used to compute "Features based Product

Table 10. Estimation accuracy of aggregation method for "features based product reputation".

Aggregation Methods	Absolute error				MAE
	P-1	P-2	P-3	P-4	
Mean+RPOAO	0.66	0.22	0.03	0.63	0.39
Mean+DPNOAO	0.66	0.22	0.17	0.49	0.39
W.Mean+RPOAO	0.56	0.16	0.04	0.45	0.30
W.Mean+DPNOAO	0.56	0.16	0.17	0.45	0.33
Proposed method	0.41	0.11	0.03	0.35	0.22

Reputation" whereas the second aggregation method (which is either RPOAO or DPNOAP) is used to compute features reputation. The results show that the average accuracy of the proposed method is better than the existing methods as the MAE is 0.22. On the other hand, the average accuracy of the Mean+RPOAO and Mean +DPNOAP are worst. The average accuracies of W.Mean+RPOAO and W.Mean +DPNOAP are slightly better than simple mean combinations. Our proposed method also offered the best accuracy in all individual products.

7. Conclusion

Huge amount of product reputation data is available on the web, which can be used to evaluate product based on features in order to determine customers' likes and dislikes. In this paper, a multi source reputation model is proposed to generate product and product feature reputation. The model offers several benefits over single source based approaches as it addresses the issue of availability and lack of ratings in single source systems. In addition, aggregation methods (such as "feature reputation", "feature-based product reputation", "product reputation", etc.) are proposed to compute different reputation values which allow the customers and manufacturers to compare products in different ways in order to make decisions. Furthermore, a method is proposed which uses four parameters (i.e., rating trustworthiness, source reliability, ageing factor and reviewer expertise) to determine rating reputability of each rating before considering for reputation value. All these rating parameters allow the model to compute reliable and trustworthy reputation values in different circumstances such as in the presence of biased ratings, malicious sources and even if there is a shift in customers' opinions. The results show that the proposed model is robust to false ratings, able to reflect the recent opinions quickly and provides a good estimation of the actual reputation value. Besides that, a balance between sensitivity and robustness is maintained which makes the model resilient to malicious users but still considering the newest ratings quickly.

In future work, the reviews will be further analyzed to produce the summary of opinions and to determine the reasons behind customers' likes and dislikes in order to provide more meaningful information to decision makers. The reputation model will be integrated with PLM (Product Life Cycle Management System) in order to allow the manufactures to make decisions throughout product life cycle.

Acknowledgment

This project has been funded with support from the European Commission (cLink Project: 372242-1-2012-1-UK-ERA MUNDUS-EMA21). This publication reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

References

1. G. Wang and K. Araki, OMS-J: An opinion mining system for Japanese Weblog reviews using a combination of supervised and unsupervised approaches, in *Proc. Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations* (Association for Computational Linguistics, 2007), pp. 19–20.
2. W. Guangwei and K. Araki, An unsupervised opinion mining approach for Japanese weblog reputation information using an improved SO-PMI algorithm, *IEICE Transactions on Information and Systems* **91**(4) (2008) 1032–1041.
3. M. Hu and B. Liu, Mining and summarizing customer reviews, in *Proc. Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (New York, USA, 2004), pp. 168–177.
4. J. Yi and W. Niblack, Sentiment mining in WebFountain, in *Proc. 21st Int. Conf. Data Engineering (ICDE 2005)*, pp. 168–177.
5. A. M. Popescu and O. Etzioni, Extracting product features and opinions from reviews, in *Natural Language Processing and Text Mining*, ed. T. Rex (Springer, London, 2007), pp. 9–28.
6. S. Morinaga, K. Yamanishi, K. Tateishi and T. Fukushima, Mining product reputations on the web, in *Proc. Eighth Int. Conf. Knowledge Discovery and Data Mining*, eds. J. Randy and K. Tan (New York, USA, 2002), pp. 341–349.
7. P. D. Thumbs Up or Thumbs Down?: Semantic Orientation applied to unsupervised classification of reviews, in *Proc. 40th Annual Meeting on Association for Computational Linguistics* (Stroudsburg, PA, USA, 2002), pp. 417–424.
8. A. Abdel-Hafez, Y. Xu and D. Tjondronegoro, Product reputation model: An opinion mining based approach, in *The 1st Int. Workshop on Sentiment Discovery from Affective Data* (Bristol, UK, 2012), pp. 16–27.
9. K. Dave, S. Lawrence and D. M. Pennock, Mining the peanut gallery: Opinion extraction and semantic classification of product reviews, in *Proc. 12th Int. Conf. World Wide Web*, 2003 (Budapest, Hungary, 2003), pp. 400–433.
10. T. D. Huynh, N. R. Jennings and N. R. Shadbolt, An integrated trust and reputation model for open multi-agent systems, *Autonomous Agents and Multi-Agent Systems* **13**(2) (2006) 119–154.
11. F. Garcin, B. Faltings and R. Jurca, Aggregating reputation feedback, in *Proc. First Int. Conf. Reputation: Theory and Technology* (Gargonza, Italy, Italian National Research Council, 2009), Vol. 9, pp. 62–74.
12. F. Garcin, B. Faltings, R. Jurca and N. Joswig, Rating aggregation in collaborative filtering systems, in *Proc. Third ACM Conf. Recommender Systems* (New York USA, ACM, 2009), pp. 349–352.
13. H. Moulin, On strategy-proofness and single peakedness, *Public Choice* **35**(4) (1980) 437–455.
14. A. Jsang, R. Ismail and C. Boyd, A survey of trust and reputation systems for online service provision, *Decision Support Systems* **43**(2) (2007) 618–644.
15. R. Antonio and P. Rosso, On the difficulty of automatically detecting irony: Beyond a simple case of negation, *Knowledge and Information Systems* **40**(3) (2014) 595–614.
16. N. Jindal and B. Liu, Opinion spam and analysis, in *Proc. 2008 Int. Conf. Web Search and Data Mining* (New York, USA, ACM, 2008), pp. 219–230.
17. E. Lim, V. Nguyen, N. Jindal, L. Nitin, L. Bing and W. Hady, Detecting product review spammers using rating behaviors, in *Proc. 19th ACM Int. Conf. Information and Knowledge Management* (Toronto, Canada, ACM, 2010), pp. 939–948.

18. D. H. Fusilier, M. Montes-y-Gomez, P. Rosso and R. G. Cabrera, Detecting positive and negative deceptive opinions using PU-learning, *Information Processing & Management* **51**(4) (2015) 433–443.
19. L. Liu and M. Munro, Systematic analysis of centralized online reputation systems, *Decision Support Systems* **52**(2) (2012) 438–449.
20. U. Farooq, Y. Ouzrout, A. Nongaillard and M. Abdul Qadir, Product reputation evaluation: The impact of conjunction on sentiment analysis, in *7th Int. Conf. Software, Knowledge, Information Management and Applications (SKIMA 2013)* (Chiang Mai Thailand, 2013), pp. 590–602.
21. J. Cho, K. Kwon and Y. Park, Q-rater: A collaborative reputation system based on source credibility theory, *Expert Systems with Applications* **36**(2) (2009) 3751–3760.
22. B. Liu, *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data* (Springer Science & Business Media, Berlin, 2007).
23. G. Song, Reputation rating mode and aggregating method of online reputation management system, *Korean Society for Industrial* **36**(2) (2009) 190–196.
24. E. K. Iverson, *APL: A Programming Language* (John Wiley & Sons, Chichester, 1962).
25. C. A. Moser and S. Alan, An experimental study of quota sampling, *Journal of the Royal Statistical Society A (General)* **116**(4) (1953) 349–405.
26. G. W. Madow and H. L. Madow, On the theory of systematic sampling, *The Annals of Mathematical Statistics* **15**(1) (1944) 1–24.
27. P. Massa and P. Avesani, Trust-aware recommender systems, in *Proc. 2007 ACM Conf. Recommender Systems* (Minneapolis, Minnesota, USA, 2007), pp. 17–24.
28. G. G. Koch, Intraclass correlation coefficient, in *Encyclopedia of Statistical Sciences* (Wiley-interscience, Berlin, 1982).
29. N. R. Maclennan, Interrater reliability with SPSS for Windows 5.0, *The American Statistician* **47**(4) (1993) 292–296.
30. G. Kou, Y. Lu, Y. Ping and Y. Shi, Evaluation of classification algorithms using MCDM and rank correlation, *International Journal of Information Technology & Decision Making* **11**(1) (2012) 197–225.