Knowl Inf Syst (2011) 26:419–434
DOI 10.1007/s10115-010-0287-y

REGULAR PAPER

An information gain-based approach for recommending useful product reviews

Richong Zhang · Thomas Tran

Received: 3 April 2009 / Revised: 7 January 2010 / Accepted: 29 January 2010 /
Published online: 3 March 2010

Abstract Recently, many e-commerce Web sites, such as Amazon.com, provide platforms for users to review products and share their opinions, in order to help consumers make their best purchase decisions. However, the quality and the level of helpfulness of different product reviews are not disclosed to consumers unless they carefully analyze an immense number of lengthy reviews. Considering the large amount of available online product reviews, this is an impossible task for any consumer. Therefore, it is of vital importance to develop recommender systems that can evaluate online product reviews effectively to recommend the most useful ones to consumers. This paper proposes an information gain-based model to predict the helpfulness of online product reviews, with the aim of suggesting the most suitable products and vendors to consumers. Reviews are analyzed and ranked by our scoring model and reviews that help consumers better than others will be found. In addition, we also compare our model with several machine learning algorithms. Our experimental results show that our approach is effective in ranking and classifying online product reviews.

Keywords Recommender systems · Product reviews · Information gain · Ranking · Scoring

1 Introduction

Recommenders are software systems that provide recommendations to assist potential buyers in choosing between diverse products and complex information. The underlying techniques based on which recommender systems are built range from personalization approach that leverages similarities between people (e.g., collaborative filtering recommenders) to approach that exploits the knowledge base of a product domain (e.g., knowledge-based recommenders), including a number of different hybrid approaches. Typically, a recommender system is

R. Zhang (✉) · T. Tran
School of Information Technology and Engineering, University of Ottawa,
800 King Edward Avenue, Ottawa, ON, K1N 6N5, Canada
e-mail: rzhan025@site.uottawa.ca

offered by an online store and only provides consumers with local recommendations about products of that store. This traditional type of recommender systems may not be very useful since consumers nowadays want to take advantages of the social web to make better informed purchase decisions by considering opinions and reviews of users from online communities and forums.

To support this trend, online reviews aggregation Web sites, such as Epinion.com, provide platforms for consumers to exchange their opinions about products, services, and merchants. "Online product reviews provided by consumers who previously purchased products have become a major information source for consumers and marketers regarding product quality" [7]. In fact, online product reviews are collectively considered as a rich source of information to help buyers make purchase decisions and are increasingly showing up as a New Genre [13]. As an illustration, Fig. 1 shows an online review page from Amazon.com. It can be seen from the figure that product reviews are sorted by "Most Helpful First", and "335 of 339 people" found that this review is helpful. There are totally 39 pages of reviews about this product. Readers are also asked to vote for this review by answering the question: " Was this review helpful to you?".

There are, however, a number of challenges for consumers to make their best use of these online reviews. As shown in the figure, reviewers write reviews and rate products by using a number of stars. Such a star scale rating, together with the fact that reviews are usually unstructured and often mix the reviewers' feelings and their opinions, makes it difficult for consumers to get the real semantics of reviews. Search engines are good tools in searching for information. However, the result set of a query returned by a search engine is typically huge. For instance, if we input '*xbox 360 reviews*' in Google, then 47,100,000 web pages will be returned to us. This massive load of information is clearly impossible for any user to handle. Also, an online community like Epinion.com usually receives more than 1,000 reviews submitted by different users for a specific product. These justify why it is essentially important to develop systems that can recommend helpful reviews to consumers effectively.

We notice that most of the review aggregation Web sites provide helpfulness voting function for consumers to rate reviews. That is, a consumer can vote a particular product review to be helpful or not helpful after he or she has read the review. Nevertheless, this progress takes time far before a really helpful review to be discovered, and the latest published review will always be the least voted one. Our goal is to filter out reviews that are most likely helpful to consumers and to provide more valuable information for consumer's decision-making process. We believe that our method can save consumers a great deal of time to surf for reliable and helpful reviews.

There is some available research focused on topical categorization, sentiment classification, and polarity identification of product reviews. In contrast, our work in this paper focuses on the modeling of consumer review helpfulness. Our model ranks reviews and returns an ordered list of reviews with the helpfulness estimates. Reviews provided by all members of the community are analyzed, and helpful opinions are presented to consumers. We examined the performance of our model on a set of reviews collected from Amazon.com. Our experimental evaluation shows that our proposed approach outperforms or performs the same as other machine learning methods.

The remainder of this paper is organized as follows: Sect. 2 discusses related work. Section 3 presents our proposed approach in details. Section 3.1 shows our experimental evaluation including the comparison between the proposed model and other machine learning methods. Section 4 provides further discussion on the value and applications of the proposed model. And finally, Sect. 5 concludes the paper and suggests some future research directions.

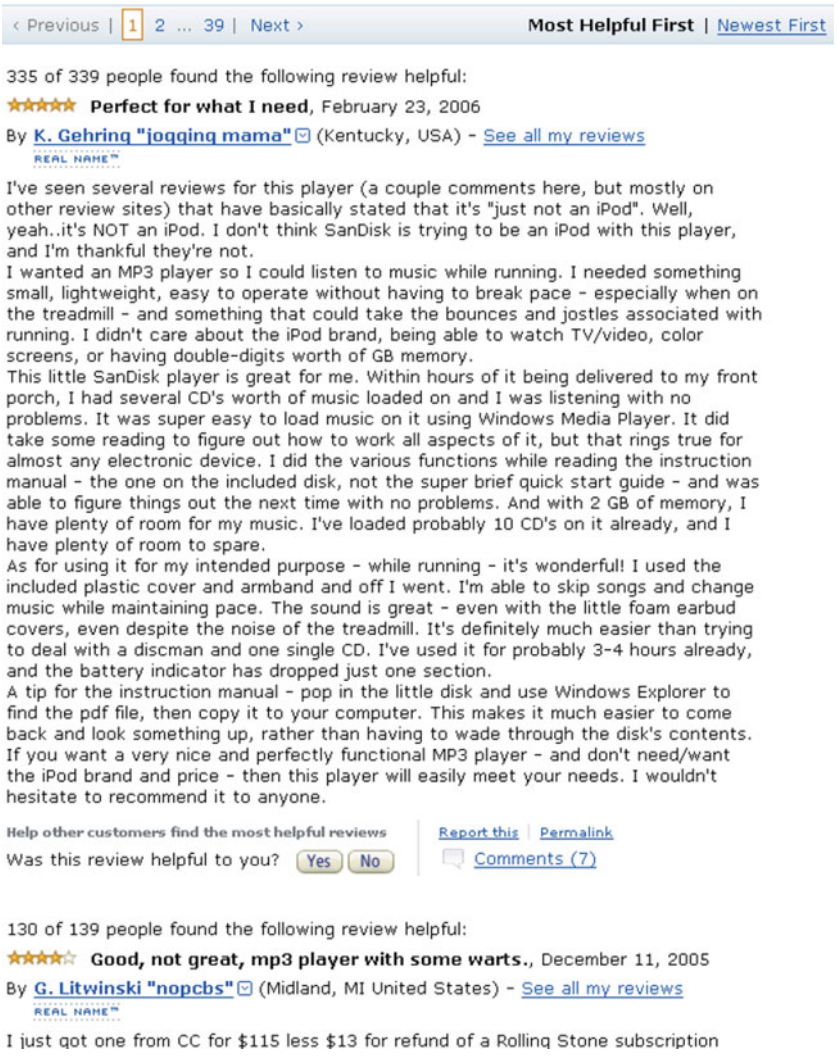


Fig. 1 An online review page

2 Related work

In order to generate appropriate recommendations and ensure the performance of recommendation systems, researchers have proposed different approaches such as collaborative filtering-based and content-based recommendation techniques. In addition, some model-based recommendation systems such as those making use of Bayesian network are also proposed. Data mining technologies, e.g., clustering and association rules etc., are introduced to build recommender systems as well.

A content-based recommender system generates recommendations based on the content of items, instead of users' opinions on these items. In this system, items are viewed as consisting of features. This method calculates the correlation between features and finds the

most relative items to recommend to users. The basic idea of the content-based method is that users prefer items similar to the ones they bought before. In general, content-based recommender systems are suitable for recommending text documents, but are not able to find similar implicit features like interests and tastes. This method is not sufficient for finding a user's potential interests.

Collaborative filtering recommendation systems predict the overall ratings by aggregating the experience of other users who are similar to the current user with respect to their interests or other aspects. The collaborative filtering recommendation approach was first proposed by David Goldberg et al. [4]. GroupLens [14] is an automatic collaborative filtering recommender system that works based on user ratings and that can be used to generate recommendations on movies, music, or news. Sarwar et al. [15] suggest that item-based collaborative filtering algorithms can perform many recommendations for millions of users and items in seconds, and the Mean Absolute Error generated by item-based collaborative filtering algorithms is lower than that generated by user-based algorithms, which indicate that item-based algorithms are able to provide higher quality recommendations.

There are also some hybrid recommender systems proposed in the literature. An example is Fab [2], which combines both content-based and collaborative filtering techniques to recommend documents. Combining these two techniques can overcome the disadvantages of using each technique alone and increase the performance of the system. Also, the model-based collaborative filtering is used to establish a model from user behaviors and generate recommendations based on this model.

The main intention of the earlier mentioned research is to generate recommendations to consumers based on different underlying approaches. However, the potential consumers neither have a chance to clearly understand why they receive such recommendations, nor have good confidence in following them. In many circumstances, consumers would like to know how other people, who have used the products that they are now interested in purchasing, rate these products.

Some researchers have been working on sentiment classification, also known as polarity classification for online product reviews, to predict whether or not consumers like a product based on the reviews. Hatzivassiloglou et al. propose a method to predict the positive or negative semantic orientation of adjectives based on a supervised learning algorithm [5]. They introduce a log-linear regression to predict the conjoined adjectives' orientation. A clustering algorithm is also introduced to group adjectives into a positive or negative class. Turney presents an unsupervised learning algorithm to classify reviews as being recommended or not recommended by analyzing their semantic orientation based on mutual information [17]. The average semantic orientation of phrases of reviews is calculated and the label of reviews is determined by this average semantic orientation. In this approach, the semantic orientations of phrases are calculated by the difference between the mutual information of the positive words and the negative words. In Ref. [23], the authors propose a classification approach to retrieve opinion sentences and separate these opinion sentences as positive or negative. In Ref. [11], the authors classify movie reviews as positive or negative by utilizing several machine learning methods, namely, Naive Bayes, Maxium Entropy and Support Vector Machines (SVM). They also make use of different features like unigram, bigram, position, and the combination of these features. Their results show that the unigram presence feature is the most effective, and the SVM performs the best for sentiment classification.

The effect of online product reviews on product sales is also a study area that recently received growing attention from researchers. In Ref. [12], the authors find that the quality of product reviews has positive effects on product sales and consumer purchase intentions increase with the quantity of good reviews. The number of online product reviews can

somehow represent the popularity of a product. Hu et al. mention that consumers not only consider review ratings but also the contextual information like the reviewer's reputation [7]. They also find that the impact of online reviews on sales diminishes over time.

Some work has been done in the area of review mining and summarizing. In Ref. [24], the authors mine and summarize the movie reviews based on a multi-knowledge approach, which includes WordNet, statistical analysis, and movie knowledge. Hu and Liu summarize product reviews by mining opinion features [6]. Wong and Lam [20] propose an approach to summarize item features and properties from multiple Web sites. Inui et al. [8] deliver a model for collecting instances of personal experiences as well as opinions from user generated contents. Under this system, consumers can perform searches for the experiences and opinions of others related to one or more topics, and the experiences returned from the system can be automatically classified into different experience classes.

Evaluating the quality and helpfulness of reviews or posts on web forums is another research domain. In Ref. [9], the authors deliver a method to automatically assess the review helpfulness. They use SVM to train their system and find that the length of the review, the unigrams and the product rating are the most important features. Weimer et al. propose an automatic algorithm to assess the quality of posts in web forum using features such as surface, lexical , syntactic, forum specific and similarity features [18]. In Ref. [19], the authors extend the method into three data sets and find that the SVM classification performs very well.

SVM has some disadvantages such as the kernel function meters have to be selected and the speed both in training and testing are all abysmally slow. In this paper, a new information gain-based approach for scoring the helpfulness of online product reviews is explored. With this approach, online product reviews can be evaluated and ranked. Then, the most useful reviews provided by other consumers will be recommended to the potential consumer based on the results of our system.

3 The proposed approach

Our work focuses on analyzing product reviews and finding high quality and helpful reviews. In this section, we first discuss how to estimate the helpfulness of reviews and define the helpfulness function. We then introduce the entropy and information gain concept, and finally, we present our proposed prediction computation.

Basically, people perform the following steps before they decide to buy a product: Perceiving a need, seeking information, comparing products, considering other users' reviews, and choosing a store for their purchase. Since users share reviews collaboratively, our system filters out useful information based on these reviews. We believe that the aggregation of community members' opinions by an effective model can generate good recommendations to help consumers with their purchase decision-making.

3.1 Review helpfulness

Consumers publish their reviews about products (or services) online after they have purchased and used the products. Normally, consumers submit their reviews to Web sites such as Epinion.com, and other potential consumers will read them. Moreover, consumers can vote a review as "Helpful" or "Not Helpful" after they read the review.

Let C be the set of consumers, P be the set of products, R be the set of reviews, and V be the set of votes that indicating the consumers' opinions about reviews (possible votes consist of "Helpful", "Not Helpful", and "Null").

- Consumer $C = \{c_1, c_2, c_3, \ldots, c_m\}$
- Product $P = \{p_1, p_2, p_3, \ldots, p_w\}$
- Review $R = \{r_1, r_2, r_3, \ldots, r_p\}$
- Vote V is a matrix listed as follows:

$$\begin{pmatrix} v_{c_1,r_1} & v_{c_1,r_2} & \cdots & v_{c_1,r_p} \\ v_{c_2,r_1} & v_{c_2,r_2} & \cdots & v_{c_2,r_p} \\ \vdots & \vdots & \vdots & \vdots \\ v_{c_m,r_1} & v_{c_m,r_2} & \cdots & v_{c_m,r_p} \end{pmatrix}$$

$$v_{c_k,r_i} = \begin{cases} Helpful & \text{if } c_k \text{ voted } r_i \text{ as Helpful,} \\ NotHelpful & \text{if } c_k \text{ voted } r_i \text{ as Not Helpful, or} \\ Null & \text{if } c_k \text{ has not voted for } r_i. \end{cases} \quad (1)$$

Definition Review helpfulness is the perception that the review $r \in R$ can be used to assist the consumers to understand the product $p \in P$. For a review r_i, its helpfulness can be calculated as the ratio of the number of consumers who have voted r_i as "Helpful" to the total number of consumers who have voted for r_i.

Let the set of all the "Helpful" votes about review r_i be denoted as h_i and the set of all "Not Helpful" votes about review r_i be denoted as $\bar{h}_i$. We define review r_i's Helpfulness as:

$$\frac{|h_i|}{|\bar{h}_i| + |h_i|} \quad (2)$$

The expected helpfulness function of an online review is a mapping $Score : R \mapsto \mathbb{R}$. The higher score a review gets, the more useful the review is.

It is clear that given an expected helpfulness function $Score$ we can sort reviews based on their scores. Furthermore, by setting a threshold θ, all reviews with score $Score > \theta$ are "Helpful" reviews, and the remaining reviews are "Not Helpful".

An online review consists of words, which include opinion words, words about product features, product parts, and other words. The importance of each word to the helpfulness of a review can be calculated from training data which contain the vote information provided by consumers.

In the following subsection, entropy and information gain will be introduced and the reason why entropy and information gain can be used to calculate the importance of words will be discussed.

3.2 Entropy and information gain

In Ref. [11], the authors reported that the best result was obtained by using Boolean values of unigram features. Motivated by this approach, we use the bag of words model to represent text and build our language model. Each feature is a non-stop stemmed word, and the value of the feature is a Boolean value of the occurrence of the word on a review.

We introduce the Shannon's information entropy concept [16] to measure the amount of information in reviews. Entropy was first developed for communication, and it has been used in many areas. It can be seen as the certainty of an event or the amount of information needed to represent an event. For the online review classification problem, the entropy can be extended as follows:

Let $S = \{s_1, s_2, \ldots, s_q\}$ be the set of categories in the online review space. The expected information needed to classify a review is:

$$H(S) = -\sum_{i=1}^{q} P_r(s_i) log\, P_r(s_i) \tag{3}$$

The average amount of information contributed by a term t in a class s_i will be:

$$H(S|t) = -\sum_{i=1}^{q} P_r(s_i|t) log\, P_r(s_i|t) \tag{4}$$

Information Gain is derived from entropy. It is originally defined as how many bits would be saved if both ends know the existence of an instance. It is often used to evaluate the relevant degree of attribute when building a decision tree [21]. In the text classification, it can be understood as the expected entropy reduction by knowing the existence of a term t. In the area of text mining and text classification, information gain is the amount of information provided by a term.

$$G(t) = H(S) - H(S|t) \tag{5}$$

Information gain is often employed as a term's goodness criterion in the field of machine learning [22]. It is used as a feature selection method in text classification and Information Retrieval deduct the dimension of documents [1,10]. In Ref. [22], information gain of term t is extended and defined as follows:

$$\begin{aligned} G(t) = &-\sum_{i=1}^{q} P_r(s_i) log\, P_r(s_i) + P_r(t) \sum_{i=1}^{q} P_r(s_i|t) log\, P_r(s_i|t) \\ &+ P_r(\bar{t}) \sum_{i=1}^{q} P_r(s_i|\bar{t}) log\, P_r(s_i|\bar{t}) \end{aligned} \tag{6}$$

In the above equations:

- $P_r(s_i)$ is the probability of documents in category s_i among all documents,
- $P_r(t)$ is the probability of documents which contain term t among all documents,
- $P_r(s_i|t)$ is the probability of documents which contain term t and which is in category s_i out of all documents containing t, and
- $P_r(s_i|\bar{t})$ is the probability of documents which do not contain term t and which belong to category s_i out of all documents which do not contain t.

The above formula calculates the reduction in entropy by knowing the occurrence of a specified term. It considers not only the term's occurrence, but also the term's non-occurrence. This value can indicate the term's contribution and predicting ability. If a word has higher information gain, it has more contribution to the classifying. For binary classification, information gain can be used to measure the amount of contribution of this term to a class.

In our case, only two categories,"Helpful" and "Not Helpful", will be considered. Let s_1 be "Not Helpful" and s_2 be "Helpful". In order to provide the difference of prediction ability for two categories, a change is introduced as follows: if $P(s_1|t) < P(s_2|t)$ then $Gain(t) = G(t)$, otherwise $Gain(t) = -G(t)$. So, the gain value calculation in our model is:

$$Gain(t) = \begin{cases} G(t) & \text{if } P(s_1|t) < P(s_2|t), \\ -G(t) & \text{otherwise.} \end{cases} \tag{7}$$

where s_1 is the category of "Not Helpful" and s_2 is the category of "Helpful".

Thus, the importance and the prediction ability of words can be calculated by Eq. (7). Table 1 shows an example of information gain values. The second and third columns are the occurring times of the specific term in the "Helpful" and "Not Helpful" domains, respectively. We will calculate the information gain values for all the terms that have occurred in the review documents.

3.3 Prediction computation

From the discussion in the above subsection, the Gain value could represent the words' ability of correctly predicting if a document belongs to "Helpful" or "Not Helpful" reviews. We use the summation of the Gain values of all words in a review to indicate the review's helpfulness. In our approach, a review's content (words) will be analyzed and the Gain value will be calculated for each word (excluding stop words) of the review. As a result, to calculate the helpfulness of a review r_i, we propose the score calculation equation as follows:

$$Score(r_i) = \sum_{j=1}^{M} Gain(t_j) * f(r_i, t_j) \tag{8}$$

where $Gain(t_j)$ is the jth stemmed word's Gain value; M is the total number of stemmed words in review r_i, and

$$f(r_i, t_j) = \begin{cases} 1 & \text{if term } t_j \text{occurrs in } r_i, \text{ or} \\ 0 & \text{if term } t_j \text{does not occurre in } r_i. \end{cases} \tag{9}$$

Equation (8) can be seen as the total helpfulness information delivered by review r_i. This equation can be used as a model to predict the helpfulness. All the score values of reviews $r_i \in R$ will be calculated. As a result, tuples of the form $< r_i, Score(r_i) >$ will be returned by this approach, where r_i is an online product review and $Score(r_i)$ is the review r_i's helpfulness value. Finally, online product reviews are ranked based on their corresponding $Score(r_i)$ values and reviews with higher score values are more helpful than others. With a set $\mathbf{T}$ of training reviews of a specific product category, and a set $\mathbf{T}'$ of test reviews, the helpfulness prediction process will be as follows:

1. Find the $Gain$ values for every non-stop word from $\mathbf{T}$.
2. Calculate the $Score$ value for every review of $\mathbf{T}'$ by Eq. (8).
3. Sort $\mathbf{T}'$ in descending order based on their $Score$ values.

Table 1 Information gain value example

term	Not Helpful	Helpful	Information gain
nuvi	55	174	0.086522889
bluetooth	11	93	0.072981165
mount	13	96	0.071278275
screen	33	118	0.055793201
crash	10	2	−0.004942425
uninstal	7	0	−0.008155578
minimum	10	0	−0.011693703

4 Experimental evaluation

In this section, we first introduce the evaluation method used in our experiments. Then, we describe the data set and the experimental steps. At the end, we analyze the experimental results and evaluate the performance of our approach for classification and ranking.

4.1 Evaluation method

In order to evaluate the performance of our model, precision and recall rate are used. Precision and recall are commonly used in evaluating information retrieval systems. Precision is defined as the ratio of retrieved helpful reviews to the total number of reviews retrieved. Recall is defined as the ratio of the number of retrieved helpful reviews to the total number of helpful reviews. "Precision and recall are thought of as some degree of correction and completeness of result" [3].

$$Precision = \frac{Reviews\ found\ and\ helpful}{Total\ reviews\ found} \tag{10}$$

$$Recall = \frac{Reviews\ found\ and\ helpful}{Total\ helpful\ reviews} \tag{11}$$

The other commonly used performance measure is F-Measure or F-Score. F-Measure is defined as the harmonic mean of the above two measures and is calculated by

$$F = \frac{2 * Precision * Recall}{Precision + Recall} \tag{12}$$

F-Measure can evaluate the overall performance of an information retrieval approach.

We also apply Spearman's rank correlation coefficient between ranks to evaluate the ranking performance of our model. The list of test reviews in descending order of the percentage of positive votes is considered as a benchmark for our experiments. Rank correlation coefficient is one of the most common methods to compare two rankings on the same set of items in statistics. If the correlation between two rankings is perfect, the value is 1. If the two rankings are totally diverse from each other, the value is -1. This method will assess the relationship between predicted rankings and real rankings. The correlation of two variables X and Y can be calculated by:

$$r = 1 - \frac{6 \sum_{i}^{n} (x_i - y_i)^2}{n(n^2 - 1)} \tag{13}$$

where n is the number of values in each set, x_i and y_i are the benchmark rank and generated ranks of the review i in the testing set respectively.

To determine the statistical significance of the Spearman's rank correlation coefficient, a t-test can be performed to test the null hypothesis of zero correlation ($r = 0$). The t-value is given by the following formula:

$$t = r\sqrt{\frac{(n-2)}{1-r^2}} \tag{14}$$

4.2 Data set

We crawled 9955 GPS and MP3 player reviews from Amazon.com. Each of the reviews has been evaluated by at least four consumers as helpful or not helpful. We define that if the

helpfulness of a review (percentage of helpful votes) is greater than 60%, the review will be marked as helpful, otherwise it is not helpful. 720 GPS reviews and 800 MP3 player reviews were randomly selected to perform the experiment.

After the parsing and stemming to all the training reviews, a document-term matrix is returned. Information gain is calculated for each term and is assigned a plus or minus sign based on the helpfulness. Thus, the document-term matrix associated with the Gain value of each stemmed word can be generated. With the Gain value, features and their corresponding importance can be discovered. We use a basic test to evaluate the performance of our method. In the basic testing stage, 300 "Helpful" GPS reviews and 300 "Not Helpful" GPS reviews are randomly chosen to form the training data set. Moreover, 60 "Helpful" GPS reviews and 60 "Not Helpful" GPS reviews are randomly selected to be utilized as the testing data set. In order to get a convincing result, we also apply 10-fold cross-validation in evaluating the performance. Reviews are randomly divided into 10 equal-sized folds. Ninefolds of the reviews are used for training the model and onefold is used as the testing data. We apply all of the 1520 GPS and MP3 online reviews in the 10-fold cross-validation.

4.3 Results and analysis

Figures 2 and 3 show the distribution of review score values from the experimental result. Most of the "Helpful" reviews and "Not Helpful" reviews concentrate on the two ends of the score value space. "Helpful" online reviews will have greater scores and "Not Helpful" online reviews will have smaller scores. This distribution highly indicates that our *Score* function can model the helpfulness of online product reviews. Therefore, with the ranking of scores, most helpful reviews can be retrieved on the top of the sorted review list.

The goal of our proposed algorithm in Sect. 3 is to calculate the helpfulness score for reviews. In order to categorize the sorted reviews into "Helpful" and "Not Helpful" with the scores returned by our algorithm, a Helpfulness Threshold is needed to be selected to build a classifier. Let N be the number of helpful reviews in the training set (where a review is said to be helpful if the percentage of helpful votes is greater than 60%, as we defined earlier). Suppose we calculate the score values of all reviews in the training set and sort the set in descending order of the score values. Then, we define the Helpfulness Threshold to be the score value of the Nth review in the sorted training set.

Table 2 is the confusion matrix of the result from the basic test. It shows the performance of our model in the basic test. In this test, 41 out of 60 "Helpful" reviews are classified as "Helpful" and 56 out of 60 "Not Helpful" reviews are classified as "Not Helpful." This is the first try to evaluate our model. Only, a small number of online reviews were involved to do

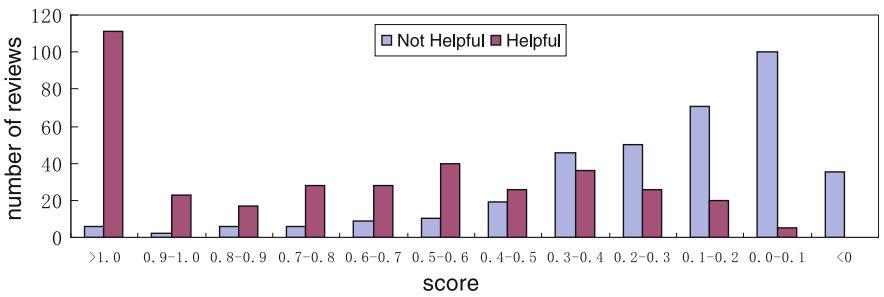


Fig. 2 Distribution of reviews' scores

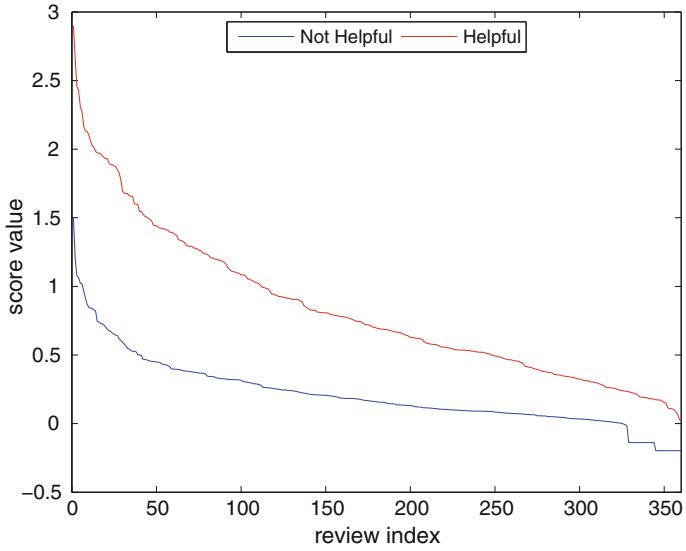


Fig. 3 Score values of reviews

Table 2 Confusion matrix (basic test)

	Not Helpful	Helpful
Training data (300 helpful reviews, 300 not helpful reviews)		
Not Helpful	234	66
Helpful	74	226
Test data (60 helpful reviews, 60 not helpful reviews)		
Not Helpful	56	4
Helpful	19	41

Table 3 Precision, recall and F-measure (10-fold cross-validation)

	Precision		Recall		F-Measure	
	MP3 Player	GPS	MP3 Player	GPS	MP3 Player	GPS
Not Helpful (%)	69.7	77	73.5	78.1	71.5	77.5
Helpful (%)	72	77.7	68	76.7	69.9	77.2

the training and testing. In order to achieve overall precision and recall, we perform 10-fold cross-validation on a bigger online review set which is described in the above subsection.

Table 3 presents the classification performance of our model by 10-fold cross-validation. The classification precision for GPS reviews is 77.7% for "Helpful" reviews and 77% for "Not Helpful" reviews. Among the MP3 Player Reviews, the precision is 69.7% for "Not Helpful" reviews and 72% for "Helpful" reviews. It shows that the classification performance of our approach is efficient to categorize the online product reviews into "Helpful" and "Not Helpful".

Table 4 Performance of various classification methods and our model for GPS reviews (10-fold cross-validation)

	Precision		Recall		F-Measure	
	Helpful	Not Helpful	Helpful	Not Helpful	Helpful	Not Helpful
Naive Bayes	0.747	0.768	0.778	0.736	0.762	0.752
SMO	0.796	0.749	0.728	0.814	0.761	0.78
Decision tree	0.77	0.714	0.681	0.797	0.723	0.753
Our model	0.777	0.77	0.767	0.781	0.772	0.775

Table 5 Performance of various classification methods and our model for MP3 reviews (10-fold cross-validation)

	Precision		Recall		F-Measure	
	Helpful	Not Helpful	Helpful	Not Helpful	Helpful	Not Helpful
Naive Bayes	0.669	0.69	0.708	0.65	0.688	0.669
SMO	0.655	0.666	0.678	0.643	0.666	0.654
Decision tree	0.611	0.619	0.633	0.598	0.622	0.608
Our model	0.697	0.72	0.735	0.68	0.715	0.70

Table 4 is the result comparison between 10-fold cross-validation performed by our model and other classification methods for GPS reviews. We compare the precision, recall, F-measure and model for Naive Bayes, Decision Tree, SMO and our information gain-based model. The experimental result reveals that our model performs at a higher or equally compatible level as other machine learning classification methods. For example, our approach outperforms Naive Bayes and Decision Tree. In comparison with SMO, our approach is 1% lower for the F-measure of "Not Helpful" reviews and 1% higher for the "Helpful" reviews. Table 5 is the result comparison between 10-fold cross-validation performed by our model and other classification methods for MP3 reviews. The experiment results show that our model outperforms the Naive Bayes, Decision Tree, and SMO algorithm. The F-measure evaluation resulted by our algorithm for Helpful reviews is at least 6.5% higher than the other three algorithms that we compared with.

To evaluate a generated ranking, the Spearman's rank correlation coefficient is adopted to estimate the correlation between the predicted ranking and the real ranking in the data set. Table 6 reports the ranking quality tests of Spearman rank order correlation coefficient with t-value. It shows the correlation between the ranking manually voted by consumers and the ranking predicted by our model and SMO regression on the two categories, namely, MP3 players and GPS in the same setting: we use the same data set and the same feature selection methods in both our model and SMO regression algorithm. Furthermore, the results are obtained through 10-fold stratified cross-validation. The rank correlation coefficient between the ranking generated by our model and the ranking voted by consumers is 0.5977 for GPS and 0.5201 for MP3 player, over the 10-fold cross-validation. The t-values of 6.4544 (for GPS) and 5.4262 (for MP3) are significant at the 0.005 probabilistic level. The coefficients and t-values presented in the table suggest a significant correlation between the predicted helpfulness ranking and the original helpfulness ranking. We also compare our model with

Table 6 Performance evaluation of our model ranking reviews of GPS and MP3 players (using 10-fold cross-validation)

Collection	Metric	SMO regression	Our model
GPS	Spearman rank order correlation	0.4953	0.5977
	t-value	4.9253	6.4544
MP3	Spearman rank order correlation	0.3831	0.5201
	t-value	3.7253	5.4262

SMO Regression. It can be seen from Table 6 that our model outperforms SMO Regression for both the GPS and MP3 reviews.

Although, in comparison with other classification and regression algorithms, our model obtains the best performance, we observe that the classification and ranking performance of our model (and also of other algorithms we compared with) for GPS reviews are better than the performance for MP3 reviews. A possible reason is that the document-term matrix of MP3 reviews is sparser and contains more noise data than the document-term matrix of GPS reviews. As a result, the information gain values of the terms in GPS reviews is greater than the information gain values of the terms in MP3 reviews, which means that the terms in GPS reviews for which a greater information gain values are available are more likely to appear in "Helpful" reviews than the terms that have greater information gain values in MP3 reviews. Accordingly, the variances of the helpfulness scores of GPS reviews are bigger than the variances of the helpfulness scores of MP3 reviews and the performances of classification and ranking for GPS reviews are than that for MP3 reviews. To mitigate this problem, some external knowledge might be introduced to find the relationship between terms.

5 Discussion

Traditional centralized recommender systems that are located at certain e-commerce stores and recommend products and services local to those stores no longer serve the increasing needs of today's consumers. Indeed, consumers in our day would like to take advantage of the social web to make better informed and more effective purchase decisions by considering the opinions of other users prior to their purchase. More and more e-commerce Web sites provide facilities for users to review products and exchange opinions. Users' product reviews have formed a rich source of information based on which satisfactory purchase decisions can be made. The information gain approach presented in this paper serves as a basic step toward the goal of recommending the most useful product reviews to consumers.

The collaborative filtering approach works well to recommend ranked products by making use of users' profiles. However, for our model, which is to recommend helpful reviews, the data for performing collaborative filtering experimentation are not available even if reviews are considered as a "product". The profiles of users who have voted for reviews are not available to us. The only available data are the review documents themselves and the number of "Helpful" and "Not Helpful" votes for each review. Our model is based on the examination of the terms in a review, which reflect more "insight" into the opinions of users on the review. Whereas the collaborative filtering approach is based on the similarity and dissimilarity between users' profiles. This difference allows our model to avoid some well-known disadvantages of the collaborative filtering approach, such as the cold start problem and the early rater problem.

Today's online customers are also known to be more impatient and demanding than ever before. On the one hand, they would like to make the best possible purchase decisions based on as much information available as possible. On the other hand, they want to complete their purchases in as little time as possible. In other words, customers want to receive satisfactory products and services, but they are not willing to spend much time and effort for their purchases. Therefore, we believe that our proposed model, aiming to automate the process of analyzing and ranking online product reviews to recommend the most helpful ones to consumers, is desirable.

We note that our proposed model can be tuned to process not only product reviews but also other sources of product and service-related information such as users' ratings, opinions and comments that can be obtained from different online user clubs, communities or forums across the Internet. It should be clear from the description of our model that it can serve as a core component in a complete and intelligent recommender system. The output of our model, i.e. the most helpful users' reviews of products or services, can be used to feed as input to the second component of the system, which will then recommend the most suitable products (or services) and vendors to consumers based on the useful reviews. In fact, such a recommender system can be configured to make recommendations of either useful reviews (or similar information), or products/services and vendors, or both, depending on the users' preferences.

From an online store's perspective, a recommender system can be developed based on our proposed model. This system would make use of user reviews and similar sources of information that are available within the store's boundaries such as the store's member clubs. From a wider and across-organization perspective, different related businesses or organizations can join together to build more versatile and useful recommender systems by using our proposed approach, which can access much larger sources of product reviews and similar information from multiple organizations as well as many available users' communities and forums across the Internet.

6 Conclusion and future work

We have proposed an information gain approach for modeling the helpfulness of online product reviews. The empirical results have shown that our model can effectively classify and rank online product reviews based on the "helpfulness score" generated by the model. Reviews with high "helpfulness scores" can then be found and recommended to consumers. Obviously, this review recommending system can be used to help consumers reach helpful reviews more easily, and a recommender system that incorporates the function of recommending useful reviews would be more useful and attract more potential buyers.

We have compared our model with other state-of-the-art algorithms for classifying and ranking, and found that our model is comparable with those algorithms in GPS and MP3 reviews. In addition, in comparison with other helpfulness assessment methods, our model is simpler and easier to understand and implement. The time complexity of our model is $O(D * W)$, where D is the number of reviews in the training set and W is the number of non-stop words in the training set. Therefore, the proposed model can classify and rank reviews significantly faster than other algorithms.

In our proposed model, we have assumed that the training set is basically from one product category and have not yet taken the issue of topic detection into account. We plan to include this in our future work to increase the accuracy of review recommendations. We only considered electronic product reviews (i.e., GPS and MP3 reviews) in this work. In

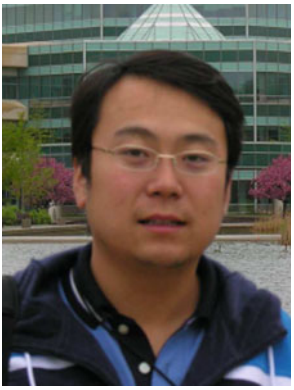
order to evaluate the performance of helpfulness reviews' recommendation in other product categories, we want to examine product reviews from different categories and explore much larger review set in the future. The decision threshold selection for classifying reviews into "Helpful" and "Hot helpful" does not employ the threshold adjusting approaches in this proposed model. It is also important to further investigate how the threshold selection methods, such as to achieve a maximized value of F-measure, may be introduced to balance the classification performance. Another future work direction would be to incorporate other factors, which may affect the quality of a review, to improve the precision of recommendations, e.g., when the review is published, how consumers rate the product, and which emotional terms and product features are used in the review. In this paper, we also assumed that all consumers have similar preferences for online reviews and did not consider the difference of individuals. Therefore, we would like to investigate as well the issue of personalization and consider the similarity and dissimilarity of consumers in our model, in order to generate personalized recommendations of helpful reviews for different consumers.

References

1. Anagnostopoulos A, Broder AZ, Punera K (2008) Effective and efficient classification on a search-engine model. Knowl Inf Syst 16(2):129–154
2. Balabanovic M (1997) An adaptive web page recommendation service. In: Proceedings of the first international conference on autonomous agents, pp 378–385
3. Euzenat J (2007) Semantic precision and recall for ontology alignment evaluation. In: Proceedings of the 2007 international joint conference on artificial intelligence, pp 348–353
4. Goldberg D, Nichols D, Oki BM, Terry D (1992) Using collaborative filtering to weave an information tapestry. Commun ACM 35(12):61–70
5. Hatzivassiloglou V, McKeown KR (1997) Predicting the semantic orientation of adjectives. In: Proceedings of the eighth conference on European chapter of the association for computational linguistics, pp 174–181
6. Hu M, Liu B (2004) Mining and summarizing customer reviews. In: Proceedings of the tenth ACM international conference on knowledge discovery and data mining, pp 168–177
7. Hu N, Liu L, Zhang J (2008) Do online reviews affect product sales? The role of reviewer characteristics and temporal effects. Inf Technol Manag
8. Inui K, Abe S, Hara K, Morita H, Sao C, Eguchi M, Sumida A, Murakami K, Matsuyoshi S (2008) Experience mining: building a large-scale database of personal experiences and opinions from Web documents. In: Proceedings of the 2008 IEEE/WIC/ACM international conference on web intelligence and intelligent agent technology, pp 314–321
9. Kim SM, Pantel P, Chklovski T, Pennacchiotti M (2006) Automatically assessing review helpfulness. In: Proceedings of the 2006 Conference on empirical methods in natural language processing, pp 423–430
10. Lee C, Lee GG (2006) Information gain and divergence-based feature selection for machine learning-based text categorization. Inform Process Manag 42
11. Pang B, Lee L, Vaithyanathan S (2002) Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL-02 conference on empirical methods in natural language processing, pp 79–86
12. Park DH, Lee J, Han I (2007) The effect of on-line consumer reviews on consumer purchasing intention: the moderating role of involvement. Int J Electron Commerce 11(4):125–148
13. Pollach I (2006) Electronic word of mouth: a genre analysis of product reviews on consumer opinion web sites. HICSS 3:1530–1605
14. Resnick P, Iacovou N, Suchak M, Bergstrom P, Riedl J (1994) GroupLens: an open architecture for collaborative filtering of netnews. In: Proceedings of the 1994 ACM conference on computer supported cooperative work, pp 175–186
15. Sarwar BM, Karypis G, Konstan JA, Reidl J (2001) Item-based collaborative filtering recommendation algorithms. World Wide Web pp 285–295
16. Shannon CE (2001) A mathematical theory of communication. SIGMOBILE Mob Comput Commun Rev 5(1):3–55

17. Turney P (2002) Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In: Proceedings of the 40th annual meeting of the association for computational linguistics (ACL), pp 417–424
18. Weimer M, Gurevych I, Mühlhäuser M (2007) Automatically assessing the post quality in online discussions on software. In: Proceedings of the 45th annual meeting of the association for computational linguistics, pp 125–128
19. Weimer M, Gurevych I (2007) Predicting the perceived quality of web forum posts. In: Proceedings of the conference on recent advances in natural language processing
20. Wong T, Lam W (2008) Learning to extract and summarize hot item features from multiple auction web sites. Knowl Inf Syst 14(2):143–160
21. Wu X, Kumar V, Quinlan J, Ghosh J, Yang Q, Motoda H, McLachlan GJ, Ng A, Liu B, Yu PS, Zhou Z, Steinbach M, Hand DJ, Steinberg D (2008) Top 10 algorithms in data mining. Knowl Inf Syst 14(1):1–37
22. Yang Y, Pedersen JO (1997) A comparative study on feature selection in text categorization. In: Proceedings of the fourteenth international conference on machine learning, pp 412–420
23. Yu H, Hatzivassiloglou V (2003) Toward answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences. In: Proceedings of the 2003 conference on empirical methods in natural language processing, pp 129–136
24. Zhuang L, Jing F, Zhu XY (2006) Movie review mining and summarization. In: Proceedings of the 15th ACM international conference on information and knowledge management, pp 43–50

Author Biographies



Richong Zhang received his B.Sc. degree and M.A.Sc. degree from Jilin University, Changchun, China, in 2001 and 2004. In 2006, he received his M.Sc. degree from Dalhousie University. He is currently a Ph.D. candidate at the School of Information Technology and Engineering, University of Ottawa. His research interests include Recommender Systems, Data Mining, and Electronic Commerce.

Thomas Tran received his Ph.D. from the University of Waterloo in 2004. He is currently an Assistant Professor at the School of Information Technology and Engineering, University of Ottawa. He is also a member of the Institute of Electrical and Electronics Engineers (IEEE), the Association for the Advancement of Artificial Intelligence (AAAI), formerly the American Association for Artificial Intelligence, and the Canadian Artificial Intelligence Association (CAIAC), formerly the Canadian Society for the Computational Studies of Intelligence. His research interests include Artificial Intelligence (AI), Electronic Commerce, Intelligent Agents and Multi-Agent Systems, Trust and Reputation Modeling, Reinforcement Learning, Recommender Systems, Knowledge-Based Systems, Architecture for Mobile E-Business, and Applications of AI.