HELPFUL OR UNHELPFUL: A LINEAR APPROACH FOR RANKING PRODUCT REVIEWS

Richong Zhang
School of Information Technology and Engineering
University of Ottawa
800 King Edward Ave., Ottawa, Ontario, K1N 6N5, Canada
rzhan025@site.uottawa.ca

Thomas Tran
School of Information Technology and Engineering
University of Ottawa
800 King Edward Ave., Ottawa, Ontario, K1N 6N5, Canada
ttran@site.uottawa.ca

ABSTRACT

Most E-commerce web sites and online communities provide interfaces and platforms for consumers to express their opinions about a specific product by writing personal reviews. The fast development of E-commerce has caused such a huge amount of online product reviews to become available to consumers that it is impossible for potential consumers to read through all the reviews and to make a quick purchasing decision. Review readers are asked to vote if a review is “Helpful” or “Unhelpful” and the most positively voted reviews are placed on the top of product review list. However, the accumulation of votes takes time for a review to be fully voted and newly published reviews are always listed at the bottom of the review list. This paper proposes a linear model to predict the helpfulness of online product reviews. Reviews can be quickly ranked and classified by our model and reviews that may help consumers better than others will be retrieved. We compare our model with several machine learning classification algorithms and our experimental results show that our approach effectively classifies online reviews. Also, we provide an evaluation measurement to judge the performance of the helpfulness modeling algorithm and the results show that the helpfulness scores predicted by our approach consistently follow the changing trend of the true helpfulness values.

Keywords: recommender system, online product review, helpfulness, evaluation

1. Introduction

Due to the fast development of Internet and E-commerce, more and more online reviews aggregation web sites, such as Epinions.com etc., have provided consumers with platforms 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” [Hu & Zhang 2008]. Park et al. [2007] confirmed that the quality of reviews has a positive effect on product sales and consumers purchase intentions increase with the quantity of product reviews.

On the E-commerce web sites, such as Amazon.com and Ebay.com, consumers are asked to write reviews and rate products or services by a number of stars after they finished a transaction. Most of existing recommendation approaches [Goldberg et al. 1992; Resnick et al. 1994; Sarwar et al. 2001] are merely based on the rating of products. With a star rating scale, users can not get `real semantics' of review statements. Since product reviews represent reviewers' feelings, experiences and opinions on a specific product, they are more useful than product ratings and therefore can better help potential consumers make purchase decisions. Search engines are good tools to assist in looking for information; however, there are too many search results returned from a search engine and not all of them are reviews. For instance, if we input `Cyber-shot Digital Camera Review' in Google, 278,000 web pages will be returned. This is absolutely a too huge result set for consumers to go through. Moreover, in an online community, such as Epinion.com or Amazon.com, more than 1000 reviews for a specific product are submitted by different consumers. Therefore, it is important to rank and classify product reviews so that they can be accessed easily and used effectively by consumers.

Review aggregation web sites provide a function for consumers to vote whether a review is “Helpful” or “Unhelpful”. But this progress takes time far before a really helpful review is discovered and the most recent published reviews will always be the least voted ones. Our goal is to develop a model that filters out reviews which are most likely helpful to consumers and that provide more valuable information for consumer's decision making. Such a model can save a great deal of consumers' effort in surfing for reliable and helpful reviews.

Most of recent researches focus on topical categorization, sentiment classification and polarity identification of consumer reviews. In this paper, we propose a linear-time model, which utilizes the helpfulness gain of each term occurring in review documents, in order to calculate the helpfulness score of product reviews. Reviews provided by all members of a community can be analyzed by our model and the helpfulness of each review can be returned by our algorithm. Moreover, we provide a metric, which utilizes the log-likelihood function of voters' opinion to evaluate the performance of the review helpfulness assessment algorithms. Not like the conventional researches only comparing the observed and the predicted helpfulness value, our evaluation measurement considers both the opinion of each voter and the number of voters who have voted. We examine the performance of our model on the reviews collected from Amazon.com and conclude that our approach can quickly and effectively discover the helpfulness of online product reviews. Our model can also be used to classify product reviews as “Helpful” or “Unhelpful”. Empirical results indicate that the classification ability of our model outperforms or performs the same as other commonly used methods in comparison with other classifiers.

The remainder of this paper is organized as follows: Section 2 discusses related work. Section 3 describes our proposed approach in details including score calculating and prediction generating. Section 4 presents our experimental evaluation which includes a comparison between the proposed model and other machine learning methods. Section 5 provides further discussion on the value and applications of the proposed model; and finally, Section 6 concludes the paper and suggests some future research directions.

2. Related Work

Park and Kim [2008] investigated the relationship between different types of reviews and consumers. They find that consumer concerns vary at each stage of the product life cycle, and suggest marketers develop different strategies for different types of consumers. Lee et al. [2008] examined the effect of the quality of negative online reviews on product attitude and discover that high-involvement consumers consider the quality of negative reviews and low-involvement consumers tend to conform to other reviewer attitudes regardless of the quality. In [Vermeulen and Seegers 2009], authors studied the effect of online hotel reviews on consumer consideration and conclude that positive reviews have a positive impact on consumer behavior. These observations indicate that online product reviews play an important role for consumers to make purchasing decisions.

Some research works have been done on sentiment classification, also known as polarity classification for online product reviews. To distinguish or predict whether consumers like a product or not based on their reviews, authors propose a method to predict the semantic orientation of adjectives by a supervised learning algorithm in [Hatzivassiloglou & McKeown 1997]. Turney presents an unsupervised learning algorithm to classify reviews as recommended or not recommended by analyzing the semantic orientation based on mutual information [Turney 2001]. In [Yu & Hatzivassiloglou 2003], the authors propose a classification approach to separate sentences as positive or negative. In [Pang et al. 2002], authors classify movie reviews as positive or negative by several machine learning methods, namely Naive Bayes, Maximum Entropy, and Support Vector Machines, and they also use different features such as unigram, bigram, position and the combination of these features. The results show that unigram presence feature was the most effective and the SVM performed the best for sentiment classification.

The effect of online product reviews on product sales is also a study area. Hu and Zhang [2008] discover that consumers consider not only the review ratings but also the contextual information like reviewer's reputation. They also find that the impact of online reviews on sales diminishes over time. Some studies have been done in the area of review mining and summarizing. In [Zhuang et al. 2006], authors mine and summarize movie reviews based on a multi-knowledge approach which includes WordNet, statistical analysis and movie knowledge. Hu and Liu [2004] summarize product reviews by mining opinion features.

Evaluating the quality and helpfulness of reviews or posts on web forums attracts more and more researchers' attentions. Kim et al. [2006] deliver a method to automatically assess review helpfulness. They use SVM to train the system and find the length of the review; the unigrams and the product rating are the most useful features. Weimer et al. [2007a] propose an automatic algorithm to assess the quality of posts on web forums using features such as surface, lexical, syntactic, forum specific and similarity features. Then the authors extend the method into the online disscussion messages on software and find the SVM classification performs very well [Weimer et al. 2007b]. Liu et al. [2008] present a nonlinear regression model for the helpfulness prediction. Three groups of factors which might affect the value of helpfulness are analyzed and the model is built upon on these three groups of factors. The results from applying their model show that the performance is better than the SVM regression model.

In this paper, we propose a linear-time helpfulness assessment approach. With this approach, online product reviews can be effectively ranked and classified, and the most helpful reviews can be retrieved to assist consumers in making purchase decisions.

3. Proposed Approach

Our work focuses on analyzing the reviews and to find high quality and helpful reviews. In this section, we discuss how to estimate the helpfulness and build the helpfulness function.

3.1. Review Helpfulness

Consumers publish their reviews about products on the review aggregation web sites or the web communities after they finish a transaction. They submit their reviews into the web site like epinion.com to share with other possible consumers. Consumers can vote reviews as “Helpful” or “Unhelpful” after they read through them.

Let *C* be the set of consumers, *P* be the set of Products, *D* be the set of review documents, and *V* be the set of votes which is consumers' opinion about reviews (vote includes “Helpful”, “Unhelpful”).

- Consumer $C = \{c_1, c_2, c_3, \ldots, c_M\}$
- Product $P = \{p_1, p_2, p_3, \ldots, p_N\}$
- Review $D = \{d_1, d_2, d_3, \ldots, d_I\}$
- Vote *V* is the set of all votes defined as follows:

We denote v_{c_m, d_i} as the consumer c_m's opinion on review document d_i. It is formulated as:

$$v_{c_m, d_i} = \begin{cases} 1 & if\ c_m\ voted\ d_i\ as\ Helpful, or \\ 0 & if\ c_m\ voted\ d_i\ as\ Unhelpful. \end{cases} \tag{1}$$

Review helpfulness is the perception that the review $d_i \in D$ can be used to assist the consumers to understand the product $p_n \in P$. For a review d_i, its helpfulness can be calculated as the ratio of the number of consumers voted d_i as “Helpful” to the total number of consumers who have voted for d_i. Let the number of all “Helpful” votes about review *d*$_i$ be denoted as $v^+_{d_i}$. Let the total number of all “Unhelpful” votes about review d$_i$ be denoted as $v^-_{d_i}$. We denote review *d*$_i$'s helpfulness as:

$$h(d_i) = \frac{v^+_{d_i}}{v^+_{d_i} + v^-_{d_i}}. \tag{2}$$

The positive vote fraction of greater than 0.9 can be seen as true helpful and smaller than 0.1 can be seen as true unhelpful. The mean of the helpfulness value of the online reviews which have a helpfulness value between 0.1 and 0.9 is 0.6. Therefore, we predefine 0.60 as a threshold for helpful reviews. If the review's helpfulness is greater than 0.60, we say it is helpful. The online review consists of words, which include opinion words, product features, product parts and other words. The importance of each word to the helpfulness of the review can be calculated from a training data which contains the vote information provided by previous review readers. In the following subsection, we define the formulation of helpfulness gain which represents the importance of word for the class of “Helpful”.

3.2. Helpfulness Gain

Pang et al. [2002] classified movie reviews as positive or negative by several machine learning methods and discovered that the best result for classifying review documents as positive or negative was obtained by using Boolean values of unigram features. We use the bag-of-words model to represent the documents and build our language model. Each feature is a non-stop stemmed word and the value of this feature is a Boolean value of the occurrence of the word on the review.

We introduce the Shannon's information entropy concept [Shannon 2001] to measure the amount of information in reviews. 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 review space. The expected information needed to classify a review is:

$$H(S) = -\sum_{i=1}^{m} \Pr(s_i) \log \Pr(s_i) \tag{3}$$

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

$$H(S \mid t) = -\sum_{i=1}^{m} \Pr(s_i \mid t) \log \Pr(s_i \mid t) \tag{4}$$

Information Gain is derived from entropy and can be understood as the expected entropy reduction by knowing the existence of a term t.

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

It is often employed as a term goodness criterion in the field of machine learning [Yang & Pedersen 1997] and is often used as a feature selection method in text classification. In [Yang and Pedersen 1997], information gain of term t was extended and defined as follows:

$$G(t) = -\sum_{i=1}^{q} P_r(s_i) \log \Pr(s_i) + \Pr(t) \sum_{i=1}^{q} \Pr(s_i \mid t) \log \Pr(s_i \mid t) + \Pr(\bar{t}) \sum_{i=1}^{q} \Pr(s_i \mid \bar{t}) \log \Pr(s_i \mid \bar{t}) \tag{6}$$

- $\Pr(s_i)$ is the probability of documents in category s_i among all documents
- $\Pr(t)$ is the probability of documents which contain term t among all documents
- $\Pr(s_i \mid t)$ is the probability of documents which contain term t and which is included in category s_i out of all documents which contain t
- $\Pr(s_i \mid \bar{t})$ is the probability of documents which do not contain term t and which belongs to category s_i out of all documents which do not contain t

The above formula calculates the reduction of 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 indicates the term's contribution and predicting ability. A word has higher helpfulness gain which means it has more contribution for classification. For a binary classification, this value can be used to measure the amount of contribution of term *t* to a class s_i.

In our case, only two categories, “Helpful” and “Unhelpful”, is considered. Let s_1 be “Unhelpful” and s_2 be “Helpful”. In order to provide the difference of prediction ability for two categories, we provide the formulation of helpfulness gain which represents a term's contribution amount to the class of “Helpful” reviews. The average helpfulness values of review documents where a term t_j occurs, denoted by $\bar{h}(D \mid t_j)$, is introduced as a factor to calculate the helpfulness gain of each term: if $P(s_1|t_j)<P(s_2|t_j)$ then $gain(t_j) = G(t_j) * \bar{h}(D \mid t_j)$, otherwise $gain(t_j) = -G(t_j) * (1 - \bar{h}(D \mid t_j))$. The helpfulness gain of a term t_j is calculated as:

$$gain(t_j) = \begin{cases} G(t_j) * \bar{h}(D \mid t_j) & \textit{if } P(s_1 \mid t_j) < P(s_2 \mid t_j), \\ -G(t_j) * (1 - \bar{h}(D \mid t_j)) & \textit{otherwise.} \end{cases} \tag{7}$$

where s_1 is the category of “Unhelpful”, s_2 is the category of “Helpful”, and $\bar{h}(D \mid t_j)$ is the mean of helpfulness value of the review documents where word t_j occurred. The helpfulness gain of term t_j, which represents the importance and the prediction ability of words, is addressed by Eq. (7).

3.3. Prediction Computation

From the discussion in previous section we understand that helpfulness gain represents a words' ability of correctly predicting a documents allocation to the category of “Helpful” or “Unhelpful” reviews. So, the summarization of the helpful gain of all words in a review indicates the review's helpfulness. In our approach, the review's content (words) will be analyzed and the helpfulness gain will be calculated for each word in product reviews. In order to predict the helpfulness of a review d_i, we propose the helpfulness score function as follows:

$$score(d_i) = \sum_{j=1}^{W} gain(t_j) * f(d_i, t_j) \tag{8}$$

Where $gain(t_j)$ is the j^{th} stemmed word's helpfulness gain and *W* is the number of stemmed non-stop words in review r_i

$$f(d_i, t_j) = \begin{cases} 1 & if\ term\, t_j\ occurs\ in\ d_i, or \\ 0 & if\ term\, t_j\ does\ not\ occur\ in\ d_i. \end{cases} \tag{9}$$

Eq. (6) can be seen as the total helpful information delivered by a review document I and we we utilize this function to model the helpfulness value of reviews. This value may be greater than 1, so we introduce a normalization factor to ensure that the calculated score value remains in the range of {0,1}. As a result, tuples of <*d*$_i$, *score(d*$_i$*)*> are returned from our algorithm. *Score(r*$_i$*)* is the review d$_i$'s predicted helpfulness score. Finally, online product reviews will be ranked based on their corresponding *score(d*$_i$*)* values. Reviews with higher score values is more helpful than others. With a set *T* of training reviews and a set T' of test reviews, the helpfulness prediction process is shown as follows:

1. Find the gain values for every non-stop word from T.
2. Calculate the helpfulness score for every review of T' by Eq. (8).
3. Normalize the helpfulness score.
4. Sort T' in descending order based on their helpfulness score.

Our approach also can be extended to a classification system by adding two more steps:

1. Find the gain values for every non-stop word from T.
2. Calculate the helpfulness score for each review in T and select a helpfulness threshold for classification.
3. Calculate the helpfulness score for every review in T' by Eq. (8).
4. Normalize the helpfulness score.
5. Sort T' in descending order based on their scores.
6. Classify review document as “Helpful” or “Unhelpful” based on whether the score of a document is greater than the threshold.

4. Experimental Evaluation

In this section, we first introduce the evaluation method used in our experiments which utilizes log-likelihood as the measurement to evaluate the performance of the helpfulness assessment for reviews. 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.

4.1. Evaluation

Precision, recall, F-score [Rijsbergen 1979] are commonly used in evaluating information retrieval systems. Precision is defined as the ratio of retrieved helpful reviews to the total number of review retrieved. Recall is defined as the ratio of the number of retrieved helpful reviews to the total number of helpful reviews. F-score is defined as the harmonic mean of above two measures and is calculated by:

$$F = \frac{2 * Precision * Recall}{Precision + Recall} \tag{10}$$

When analyzing the predictive success of helpfulness assessment algorithms, there exist various accepted evaluation metrics. Correlation coefficient between the predicted helpfulness ranking and the observed helpfulness ranking is often used to evaluate the performance of helpfulness assessing models. Also, other evaluation metrics commonly used in learning ranking area, such as precision at position n, mean average precision and normalized discount cumulative Gain, can be used to evaluate the helpfulness assessing systems. These evaluation metrics do not take the voters population size into account. In order to precisely justify the goodness-of-fit and the ranking performance of our model, we bring forward the log-likelihood of voter's opinion as “Helpful” to evaluate the difference between the predicted helpfulness score and the real observed helpfulness value from the data set.

With respect of a fixed set of voters which can be observed in the training set, we can assume the helpfulness value is a fixed number of the data set. Given a fixed helpfulness value, if we take a voter c_m randomly from the voters' population on a specific review document d_i, the vote or opinion of this randomly selected voter about the review document d_i can be modeled as an independent, identically distributed (i.i.d.) Bernoulli random variable governed by the population parameter and the distribuiton of voter opinions can be formulized as:

$$p(v_{c_m,d_i}) = \begin{cases} h(d_i), & if\ v_{c_m,d_i} = 1; \\ 1 - h(d_i), & else. \end{cases} \tag{11}$$

Then the probability of a voter will vote “Helpful” for a review document d_i is $p^*(v_{c_m,d_i}=1)=h(d_i)^{v^+_{d_i}}(1-h(d_i))^{v^-_{d_i}}$. Therefore, we can use this value as the benchmark to compare with the $p(v_{c_m,d_i}=1)$ of predicted helpfulness score:

$$p(v_{c_m,d_i}=1)=score(d_i)^{v^+_{d_i}}(1-score(d_i))^{v^-_{d_i}}, \tag{12}$$

where *score(d$_i$)* is the predicted helpfulness score. In the following sections, we investigate the performance of our algorithm by comparing the log-likelihood of the observed and predicted voters' opinion.

4.2. Data set

Our experiments focus on the product categories of digital cameras. We crawled 7054 digital camera reviews from Amazon.com, of which 1468 review documents have been evaluated by at least 5 consumers as helpful or unhelpful. We delete a list of 571 stop words [Buckley 1985] and use the bag-of-word model to represent text and build our language model. We apply Porter Stemming algorithm [Porter 1980] to all the words in our data set and each feature is a non-stop stemmed word and the value of this feature is a Boolean value of the occurrence of the word on the review. After the parsing and stemming to all the reviews, a document term matrix is returned associated with the helpfulness value of each document. In this data set, 11818 non-stop stemmed words are detected. We make use of 1468 review documents which have been voted by more than 5 voters and make use of the vocabulary of words or terms as the feature set of the model to build a document-term matrix and each the matrix has been normalized to zero-mean. 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 “Unhelpful”. We randomly select 600 digital camera reviews to execute the experiments. We use 10-fold cross validation to evaluate our approach. Reviews are randomly divided into 10 equal-sized folds, of which 9 folds of reviews are used for training the model and one fold used as test data.

4.3. Results and Analysis

We choose the helpfulness score of the $|V^+|^{th}$ (the number of helpful reviews in the training set) sorted review as the helpfulness which is used to classify reviews. Figure 1 and figure 2 show the resulted helpfulness score of the training set and the testing set from one of the 10-folds evaluations. Figure 1 shows the training reviews' helpfulness scores resulted from our algorithm.

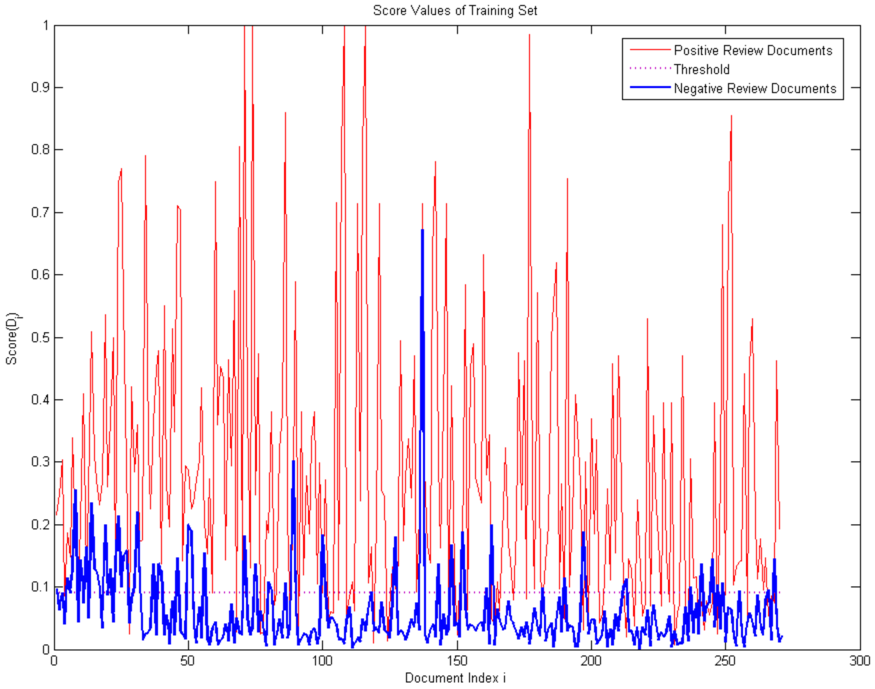

Figure 1: Score Values of Reviews in the Training Set

Clearly, most of the score values of the helpful data are greater than the threshold and most of the scores of the unhelpful review documents are smaller than the threshold. In other words, most of the “Helpful” reviews have larger helpfulness scores than “Unhelpful”. This result highly indicates that the helpfulness score function can model the helpfulness of online product reviews. So, with the ranking of helpfulness scores, most of the helpful reviews can be retrieved on the top of the sorted reviews' list.

Figure 2 shows the score values of the testing data which contains 30 “Helpful” reviews and 30 “Unhelpful” reviews. It illustrates that most of the testing “Helpful” reviews' predicted helpfulness scores are greater than the threshold which is learned from the training set, and most of the testing “Unhelpful” reviews' predicted helpfulness scores are smaller than the threshold. This clear distinction indicates that our approach can correctly group the product reviews into “Helpful” and “Unhelpful”.

In order to justify how the predicted helpfulness score differs from the true helpfulness value of the data set and the goodness-of-fit of our model, we first sort review documents with decreasing log-likelihood of $p^*(v_{c_m,d_i}=1)$, then we plot the log-likelihood of $p^*(v_{c_m,d_i}=1)$ from the data set and the log-likelihood of $p(v_{c_m,d_i}=1)$ resulted from our model for each review in the sorted reviews list. Figure 3 shows the performance of our algorithm working with “Helpful” Reviews, and Figure 4 shows the performance of our algorithm dealing with “Unhelpful” Reviews. These two figures are achieved from one of the 10-fold cross validations which includes 540 training review documents (270 “Helpful” reviews and 270 “Unhelpful” reviews) and 60 testing review documents (30 “Helpful” reviews and 30 “Unhelpful” reviews) in each fold to evaluate our approach. The experimental results of the Helpful and Unhelpful reviews have the same pattern and exhibit the same decreasing trend as the original helpfulness value of the data set. It highly indicates that our approach can effectively predict the helpfulness of the review documents.

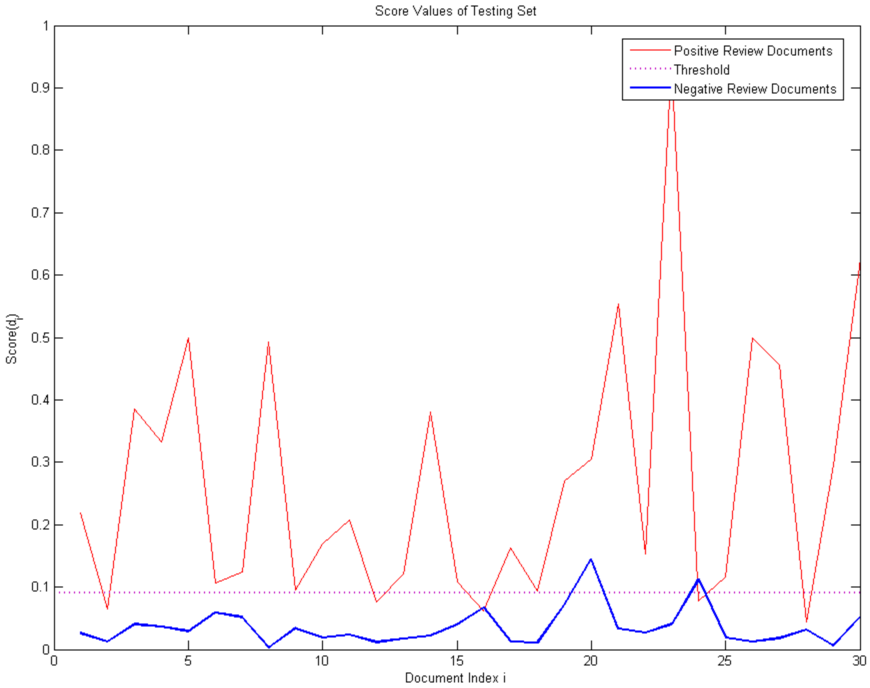

Figure 2: Score Values of Reviews in the Testing Set

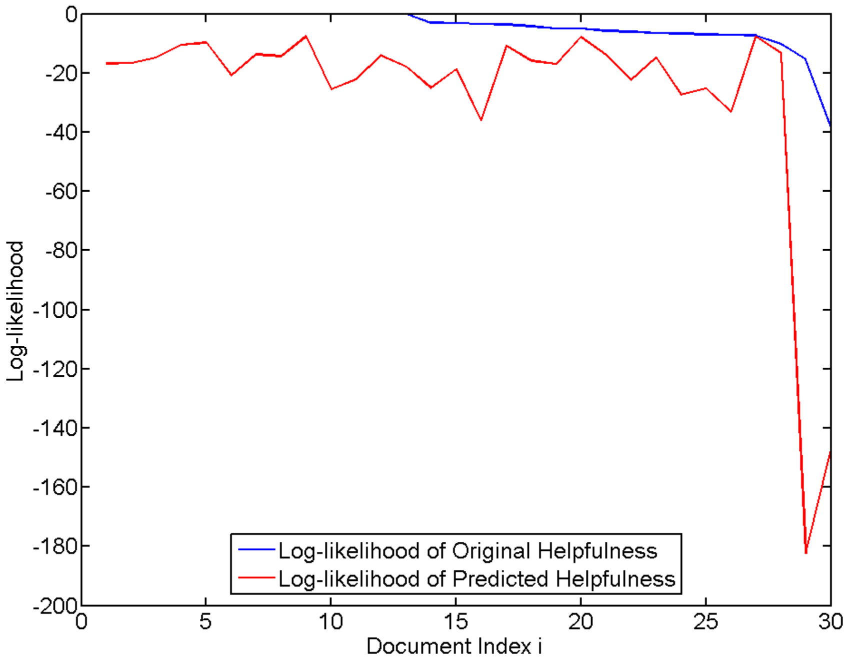

Figure 3: Log-likelihood of Positive Reviews

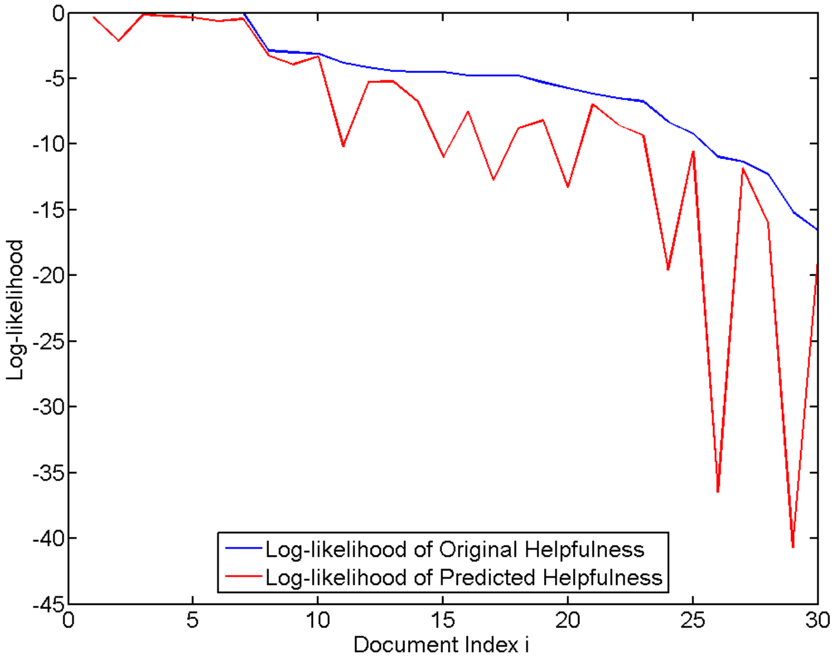

Figure 4: Log-likelihood of Negative Reviews

Table 1 shows the classification performance of our model. We use the 10-fold cross validation, which makes our results less prone to random variation, and the classification precision of our model is 76.7% for “Helpful” reviews and 73.7% for “Unhelpful” reviews. Precision, recall, and F-score of Naive Bayes, Decision Tree, SMO and our model is compared in Table 1. For both the “Helpful” and “Unhelpful” reviews, the precision and recall of our approach outperforms Naive Bayes and Decision Tree. In comparison with SMO, the recall of our approach is 3.7% higher than the recall of "Helpful" reviews and the precision is 3.3% lower than the precision of SMO method.

The experimental result led us to conclude that the classification capability our model performs better or at least the same as other commonly used classifiers.

Table 1: Performance of various classification methods and our model (10-fold cross-validation)

	Precision		Recall		F-measure	
	Helpful	Unhelpful	Helpful	Unhelpful	Helpful	Unhelpful
Naive Bayes	0.759	0.748	0.743	0.763	0.751	0.756
SMO	0.80	0.75	0.722	0.82	0.762	0.783
DecisionTree	0.717	0.694	0.696	0.713	0.733	0.733
Our Model	0.767	0.737	0.759	0.759	0.763	0.748

5. Discussion

The model discussed in this paper analyzes the helpfulness gain of each word from the training set which is calculated from the information gain of a term and the average helpfulness value of the review documents where this word occurs. With this helpfulness gain, features' contribution to the class of "Helpful" is discovered. The empirical results show that our model performs very well for our data set of Amazon.com.

We introduce the log-likelihood function of the voter's opinion as a metric to evaluate the performance of helpfulness assessment in this paper. We compare the predicted helpfulness score and the helpfulness value of the data set by this evaluation measurement. The experimental results show that the predicted helpfulness score calculated by our algorithm consistently follow the changing trend of helpfulness value of the data set. With the help of our approach, consumers can easily assess the "Helpful" reviews and don't have to spend effort searching by themselves.

In comparison with other classification approaches, the evaluation results indicate that our approach performs better or the same as other common used machine learning methods. However, we make use of a linear summarization approach to model the helpfulness of reviews which means that the time complexity of our approach is linear and is lower than other machine learning approaches.

It has been observed that the precision of our model is better for Helpful review documents than Unhelpful ones. The most likely reason is that our algorithm find more terms with higher gain value in the helpful category than in the unhelpful one.

In this paper, the experiments were run for helpfulness discovery using only unigrams. Word n-gram based text representation can also be used to represent online review documents. The performance of involving word n-gram features should be improved by involving more features. The configuration of our algorithm is very simple, and other corpora can be easily used for the helpfulness evaluation.

The threshold we predefined for separating "Helpful" and "Unhelpful" is 0.6. This value will affect the performance of our model applying to other categories of online product reviews. A threshold choosing measurement, as to achieve a maximized value of F-measure, may be introduced to balance the classification performance.

Many online shopping web sites provide facilities for consumers to share their experience and review product. This forms a huge information source for online users and it becomes more and more difficult to easily compare reviews and make decisions. Our model can serve as a review recommendation system to provide the most helpful reviews to potential consumers. Moreover, 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 can be fitted as the input to our model; and either useful reviews (or similar information), or products/services and vendors are the output of our model. Obviously, a recommender system that incorporates the function of recommending useful reviews would be much more helpful, convenient and surely attract more potential buyers.

6. Conclusion and Future Work

This paper proposes a model for modeling the helpfulness of online product reviews based on the unigram features. Using our model, online product reviews can be classified and ranked based on the score values and the most helpful reviews can be identified. This helps consumers complete their information search and make purchase decisions easily and quick. We utilize the helpfulness gain, which represents the classification capability of each term, to model the helpfulness value of reviews. In comparison with other helpfulness assessment methods, our model is simpler, easier to implement and more understandable. The linear-time complexity of our model indicates the helpfulness model can be learned quickly. We also proposed a new evaluation measurement to judge the performance of helpfulness assessment algorithms which not only compares the difference between the observed and predicted values, but also takes the voters' population size into account. Moreover, the experimental results show that our approach achieves good performance in ranking and classifying review documents.

In this paper, we have assumed that all consumers have the same preferences for online reviews and do not consider the difference between individuals. In the future research, in order to improve the accuracy of personalization, the similarity of voters should be considered in our model and more data would be collected to examine the generalization power of our model. Our intended future research would also take other feature sets, which may affect the quality of reviews, into consideration. These include such features as when a review was published, how the consumers rated the product, and the number of features which were mentioned in the review.

In our proposed model, we did not detect the intelligent spamming reviews which only consist of the words which have high gain values. We are going to include this function in our future works. The experimental results led us to conclude that our approach is able to efficiently predict the helpfulness of specific categories of review documents. Approaches that distinguish relevant and non-relevant items [Butler et. al 2001] might help us to extend our model to a more general case that can discover a helpful online review which is relevant to the item which users are interested in.

REFERENCES

Buckley, C., “Implementation of the Smart Information Retrieval System,” Technical Report TR85-686, Dept. of Computer Science, Cornell Univ., May 1985.

Butler, J., D.J. Morrice, P.W. Mullarkey, “A multiple attribute utility theory approach to ranking and selection” , In: Management Science 47(6), pp. 800-816. 2001

Goldberg, D., D. Nichols, B.M. Oki, and D. Terry, “Using collaborative filtering to weave an information tapestry,” Communication. ACM, 35(12):61-70. 1992

Hatzivassiloglou, V. and K. R. McKeown, “Predicting the semantic orientation of adjectives,” In Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics, pages 174-181, Morristown, NJ, USA. Association for Computational Linguistics. 1997

Hu, M. and B. Liu, “Mining and summarizing customer reviews.” In KDD '04: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, pages 168-177, New York, NY, USA. ACM. 2004

Kim, S.M., P. Pantel, T. Chklovski, and M. Pennacchiotti. “Automatically assessing review helpfulness,” In Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, pages 423-430, Sydney, Australia. Association for Computational Linguistics. 2006

Lee, J., D.H. Park, and T. Han, “The effect of negative online consumer reviews on product attitude: An information processing view.” Electronic Commerce Research and Applications 7, 3, 341 -352. 2008

Liu, Y., X. Huang, A. An, and X. Yu, “Modeling and predicting the helpfulness of online reviews,” In ICDM '08: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, pages 443-452, Washington, DC, USA. IEEE Computer Society. 2008

Hu, N., L. L. and Zhang, J. J. “Do online reviews affect product sales?” Information Technology and Management. 2008

Orkin, M. and Drogin, R. “Vital Statistics, McGraw-Hill,” 1990

Pang, B., Lee, L., and Vaithyanathan, S. “Thumbs up? sentiment classification using machine learning techniques,” In EMNLP '02: Proceedings of the ACL-02 conference on Empirical methods in natural language processing, pages 79-86, Morristown, NJ, USA. Association for Computational Linguistics. 2002

Park, D.H. and S. Kim, “The effects of consumer knowledge on message processing of electronic word-of-mouth via online consumer reviews.,” Electronic Commerce Research and Applications 7, 4, 399 - 410.2008

Park, D.H., J. Lee, and I. Han, “The effect of on-line consumer reviews on consumer purchasing intention: The moderating role of involvement,” Int. J. Electron. Commerce, 11(4):125-148. 2007

Porter, M.F., “An algorithm for suffix stripping”, Program, 14(3) pp 130−137.1980

Resnick, P., N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, “Grouplens: an open architecture for collaborative filtering of netnews,” In CSCW '94: Proceedings of the 1994 ACM conference on Computer supported cooperative work, pages 175-186, New York, NY, USA. ACM Press. 1994

Sarwar, B. M., G. Karypis, J.A. Konstan, and J. Reidl, “Item-based collaborative filtering recommendation algorithms.” In World Wide Web, pages 285-295. 2001

Shannon, C. E. “A mathematical theory of communication,” SIGMOBILE Mob.Comput. Commun. Rev., 5(1):3-55. 2001

Turney, P. D. “Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews,” In ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pages 417-424, Morristown, NJ, USA. Association for Computational Linguistics. 2002

Rijsbergen, C. J. “Information Retrieval,” Dept. of Computer Science, University of Glasgow, Second edition. 1997

Vermeulen, I.E. and D. Seegers, “Tried and tested: The impact of online hotel reviews on consumer consideration,” Tourism Management 30, 1, 123 - 127. 2009

Weimer, M and I. Gurevych, “Predicting the perceived quality of web forum posts,” Proceedings of the Conference on Recent Advances in Natural Language Processing (RANLP). 2007a

Weimer, M., I. Gurevych, and M. Muhlhauser, “Automatically assessing the post quality in online discussions on software,” In Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions, pages 125-128, Prague, Czech Republic. Association for Computational Linguistics. 2007b

Yang, Y. and J.O. Pedersen, “A comparative study on feature selection in text categorization,” In ICML '97: Proceedings of the Fourteenth International Conference on Machine Learning, pages 412-420, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc. 1997

Yu, H. and V. Hatzivassiloglou, “Towards 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, pages 129-136, Morristown, NJ, USA. Association for Computational Linguistics. 2003

Zhuang, L., F. Jing, and X.Y. Zhu, “Movie review mining and summarization,” In CIKM '06: Proceedings of the 15th ACM international conference on Information and knowledge management, pages 43-50, New York, NY, USA. ACM.2006