

# Topic Extraction from Online Reviews for Classification and Recommendation\*

Ruihai Dong, Markus Schaal, Michael P. O'Mahony, Barry Smyth

CLARITY: Centre for Sensor Web Technologies

School of Computer Science and Informatics

University College Dublin, Ireland

{firstname.lastname}@ucd.ie

## Abstract

Automatically identifying informative reviews is increasingly important given the rapid growth of user generated reviews on sites like Amazon and TripAdvisor. In this paper, we describe and evaluate techniques for identifying and recommending helpful product reviews using a combination of review features, including topical and sentiment information, mined from a review corpus.

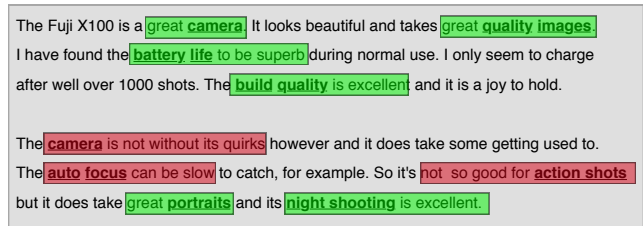
## 1 Introduction

The web is awash with user generated reviews, from the contemplative literary critiques of GoodReads to the flame wars that can sometimes erupt around hotels on TripAdvisor. User generated reviews are now an important part of how we inform opinions when we make decisions to travel and shop. The availability of reviews helps shoppers to choose [Hu *et al.*, 2008] and increases the likelihood that they will make a buying decision [Zhu and Zhang, 2010]. But the helpfulness of user reviews depends on their quality (detail, objectivity, readability etc.) As the volume of online reviews grows it is becoming increasingly important to provide users with the tools to filter hundreds of opinions about a given product.

Sorting reviews by helpfulness is one approach but it takes time to accumulate helpfulness feedback and more recent reviews are often disadvantaged until they have accumulated a minimum amount of feedback. One way to address this is to develop techniques for automatically assessing the helpfulness of reviews. This has been attempted in the past with varying degrees of success [O'Mahony *et al.*, 2009] by learning classifiers using review features based on the *readability* of the text, the *reputation* of the reviewer, the *star rating* of the review, and various *content* features based on the review terms.

In this paper we build on this research in a number of ways. We describe a technique to extract interesting topics from reviews and assign sentiment labels to these topics. Figure 1 provides a simple example based on a review of a camera. In this case the reviewer has mentioned certain topics such as

*build quality* and *action shots* in a positive or negative context. Our aim is to automatically mine these types of topics from the raw review text and to automatically assign sentiment labels to the relevant topics and review elements. We describe and evaluate how such features can be used to predict review quality (helpfulness). Further we show how this can be used as the basis of a review recommendation system to automatically recommend high quality reviews even in the absence of any explicit helpfulness feedback.



The Fuji X100 is a **great camera**. It looks beautiful and takes **great quality images**. I have found the **battery life to be superb** during normal use. I only seem to charge after well over 1000 shots. The **build quality is excellent** and it is a joy to hold.

The **camera is not without its quirks** however and it does take some getting used to. The **auto focus can be slow** to catch, for example. So it's **not so good for action shots** but it does take **great portraits** and its **night shooting is excellent**.

Figure 1: A product review for a digital camera with topics marked as bold, underlined text and sentiment highlighted as either a green (positive) or red (negative) background.

## 2 Related Work

Recent research highlights how online product reviews can influence on the purchasing behaviour of users; see [Hu *et al.*, 2008; Zhu and Zhang, 2010]. The effect of consumer reviews on book sales on Amazon.com and Barnesandnoble.com [Chevalier and Dina Mayzlin, 2006] shows that the relative sales of books on a site correlates closely with positive review sentiment; although interestingly, there was insufficient evidence to conclude that retailers themselves benefit from making product reviews available to consumers; see also the work of [Dhar and Chang, 2009] and [Dellarocas *et al.*, 2007] for music and movie sales, respectively. But as review volume has grown retailers need to develop ways to help users find high quality reviews for products of interest and to avoid malicious or biased reviews. This has led to a body of research focused on classifying or predicting review helpfulness and also research on detecting so-called *spam reviews*.

A classical review classification approach, proposed by [Kim *et al.*, 2006], considered features relating to the ratings, structural, syntactic, and semantic properties of reviews

\*This work is supported by Science Foundation Ireland under grant 07/CE/I1147.

to find ratings and review length among the most discriminating. Reviewer expertise was found to be a useful predictor of review helpfulness by [Liu *et al.*, 2008], confirming, in this case, the intuition that people interested in a certain genre of movies are likely to pen high quality reviews for similar genre movies. Review timeliness was also found to be important since review helpfulness declined as time went by. Furthermore, opinion sentiment has been mined from user reviews to predict ratings and helpfulness in services such as TripAdvisor by the likes of [Baccianella *et al.*, 2009; Hsu *et al.*, 2009; O’Mahony *et al.*, 2009; O’Mahony and Smyth, 2009].

Just as it is useful to automate the filtering of helpful reviews it is also important to weed out malicious or biased reviews. These reviews can be well written and informative and so appear to be helpful. However these reviews often adopt a biased perspective that is designed to help or hinder sales of the target product [Lim *et al.*, 2010]. [Li *et al.*, 2011] describe a machine learning approach to spam detection that is enhanced by information about the spammer’s identity as part of a two-tier co-learning approach. On a related topic, [O’Callaghan *et al.*, 2012] use network analysis techniques to identify recurring spam in user generated comments associated with YouTube videos by identifying discriminating comment *motifs* that are indicative of spambots.

In this paper we extend the related work in this area by considering novel review classification features. We describe techniques for mining topical and sentiment features from user generated reviews and demonstrate their ability to boost classification accuracy.

### 3 Topic Extraction and Sentiment Analysis

For the purpose of this work our focus is on mining topics from user-generated product reviews and assigning sentiment to these topics on a per review basis. Before we describe how this topical and sentiment information can be used as novel classification features, we will outline how we automatically extract topics and assign sentiment as per Figure 2.

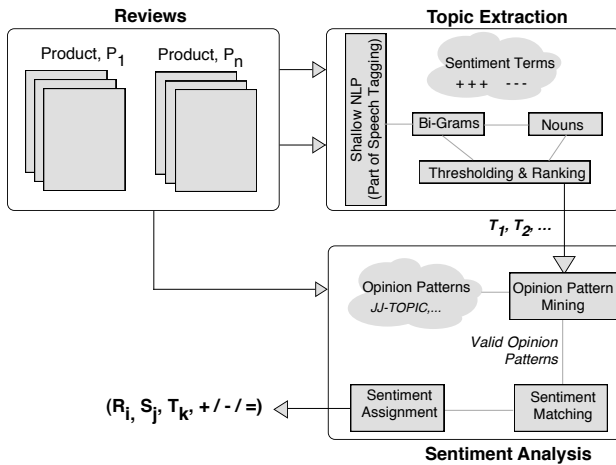


Figure 2: System architecture for extracting topics and assigning sentiment for user generated reviews.

#### 3.1 Topic Extraction

We consider two basic types of topics — *bi-grams* and *single nouns* — which are extracted using a combination of shallow NLP and statistical methods, primarily by combining ideas from [Hu and Liu, 2004a] and [Justeson and Katz, 1995]. To produce a set of bi-gram topics we extract all bi-grams from the global review set which conform to one of two basic part-of-speech co-location patterns: (1) an adjective followed by a noun (*AN*) such as *wide angle*; and (2) a noun followed by a noun (*NN*) such as *video mode*. These are candidate topics that need to be filtered to avoid including *AN*’s that are actually opinionated single-noun topics; for example, *excellent lens* is a single-noun topic (*lens*) and not a bi-gram topic. To do this we exclude bi-grams whose adjective is found to be a sentiment word (e.g. *excellent*, *good*, *great*, *lovely*, *terrible*, *horrible* etc.) using the sentiment lexicon proposed in [Hu and Liu, 2004b].

To identify the single-noun topics we extract a candidate set of (non stop-word) nouns from the global review set. Often these single-noun candidates will not make for good topics; for example, they might include words such as *family* or *day* or *vacation*. [Qiu *et al.*, 2009] proposed a solution for validating such topics by eliminating those that are rarely associated with opinionated words. The intuition is that nouns that frequently occur in reviews and that are frequently associated with sentiment rich, opinion laden words are likely to be product topics that the reviewer is writing about, and therefore good topics. Thus, for each candidate single-noun, we calculate how frequently it appears with nearby words from a list of sentiment words (again, as above, we use Hu and Liu’s sentiment lexicon), keeping the single-noun only if this frequency is greater than some threshold (in this case 70%).

The result is a set of bi-gram and single-noun topics which we further filter based on their frequency of occurrence in the review set, keeping only those topics ( $T_1, \dots, T_m$ ) that occur in at least  $k$  reviews out of the total number of  $n$  reviews; in this case, for bi-gram topics we set  $k_{bg} = n/20$  and for single noun topics we set  $k_{sn} = 10 \times k_{bg}$ .

#### 3.2 Sentiment Analysis

To determine the sentiment of the topics in the product topic set we use a method similar to the *opinion pattern mining* technique proposed by [Moghaddam and Ester, 2010] for extracting opinions from unstructured product reviews. Once again we use the sentiment lexicon from [Hu and Liu, 2004b] as the basis for this analysis. For a given topic  $T_i$ , and corresponding review sentence  $S_j$  from review  $R_k$  (that is the sentence in  $R_k$  that includes  $T_i$ ), we determine whether there are any sentiment words in  $S_j$ . If there are not then this topic is marked as *neutral*, from a sentiment perspective. If there are sentiment words ( $w_1, w_2, \dots$ ) then we identify that word ( $w_{min}$ ) which has the minimum word-distance to  $T_i$ .

Next we determine the part-of-speech tags for  $w_{min}$ ,  $T_i$  and any words that occur between  $w_{min}$  and  $T_i$ . The POS sequence corresponds to an opinion pattern. For example, in the case of the bi-gram topic *noise reduction* and the review sentence “...this camera has great noise reduction...”,  $w_{min}$  is the word “great” which corresponds to the opinion pattern *JJ-TOPIC* as per [Moghaddam and Ester, 2010].

Once an entire pass of all topics has been completed we can compute the frequency of all opinion patterns that have been recorded. A pattern is deemed to be valid (from the perspective of our ability to assign sentiment) if it occurs more than the average number of occurrences over all patterns [Moghadam and Ester, 2010]. For valid patterns we assign sentiment based on the sentiment of  $w_{min}$  and subject to whether  $S_j$  contains any negation terms within a 4-word-distance of  $w_{min}$ . If there are no such negation terms then the sentiment assigned to  $T_i$  in  $S_j$  is that of the sentiment word in the sentiment lexicon. If there is a negation word then this sentiment is reversed. If an opinion pattern is deemed not to be valid (based on its frequency) then we assign a *neutral* sentiment to each of its occurrences within the review set.

## 4 Classifying Helpful Reviews

In the previous section we described our approach for automatically mining topics ( $T_1, \dots, T_m$ ) from review texts and assigning sentiment values to them. Now we can associate each review  $R_i$  with *sentiment tuples*,  $(R_i, S_j, T_k, + / - / =)$ , corresponding to a sentence  $S_j$  containing topic  $T_k$  with a sentiment value positive (+), negative (-), or neutral (=).

To build a classifier for predicting review helpfulness we adopt a supervised machine learning approach. In the data that is available to us each review has a helpfulness score that reflects the percentage of positive votes that it has received, if any. In this work we label a review as *helpful* if and only if it has a helpfulness score in excess of 0.75. All other reviews are labeled as *unhelpful*; thus we adopt a similar approach to that described by [O’Mahony and Smyth, 2009].

To represent review instances we rely on a standard feature-based encoding using a set of 7 different types of features including temporal information (*AGE*), rating information (*RAT*), simple sentence and word counts (*SIZE*), topical coverage (*TOP*), sentiment information (*SENT*), readability metrics (*READ*), and content made up of the top 50 most popular topics extracted from the reviews (*CNT*). These different types, and the corresponding individual features are summarised in Table 1. Some of these features, such as rating, word and sentence length, date and readability have been considered in previous work [Kim *et al.*, 2006; Liu *et al.*, 2008; O’Mahony and Smyth, 2010] and reflect best practice in the field of review classification. But the topical and sentiment features (explained in detail below) are novel, and in this paper our comparison of the performance of the different feature sets is intended to demonstrate the efficacy of our new features (in isolation and combination) in comparison to classical benchmarks across a common dataset and experimental setup.

### 4.1 From Topics and Sentiment to Classification Features

For each review  $R_k$ , we assign a collection of topics ( $topics(R_k) = T_1, T_2, \dots, T_m$ ) and corresponding sentiment scores (*pos/neg/neutral*) which can be considered in isolation and/or in aggregate as the basis for classification features. For example, we can encode information about a review’s *breadth* (see Equation 1) and *depth* of topic coverage

by simply counting the number of topics contained within the review and the average word count associated with the corresponding review sentences, as in Equation 2. Similarly we can aggregate the popularity of review topics, relative to the topics across the product as a whole as in Equation 3 (with  $rank(T_i)$  as a topic’s popularity rank for the product and  $UniqueTopics(R_k)$  as the set of unique topics in a review); so if a review covers many popular topics then it receives a higher score than if it covers fewer rare topics.

$$Breadth(R_k) = |topics(R_k)| \quad (1)$$

$$Depth(R_k) = \frac{\sum_{\forall T_i \in topics(R_k)} len(sentence(R_k, T_i))}{Breadth(R_k)} \quad (2)$$

$$TopicRank(R_k) = \sum_{\forall T_i \in UniqueTopics(R_k)} \frac{1}{rank(T_i)} \quad (3)$$

When it comes to sentiment we can formulate a variety of classification features from the number of positive (*NumPos* and *NumUPos*), negative (*NumNeg* and *NumUNeg*) and neutral (*NumNeutral* and *NumUNeutral*) topics (total and unique) in a review, to the rank-weighted number of positive (*WPos*), negative (*WNeg*), and neutral (*WNeutral*) topics, to the relative sentiment, positive (*RelUPos*), negative (*RelUNeg*), or neutral (*RelUNeutral*), of a review’s topics; see Table 1.

We also include a measure of the relative *density* of opinionated (non-neutral sentiment) topics in a review (see Equation 4) and a relative measure of the difference between the overall review sentiment and the user’s normalized product rating, i.e.  $SignedRatingDiff(R_k) = RelUPos(R_k) - NormUserRating(R_k)$ ; we also compute an unsigned version of this metric. The intuition behind the rating difference metrics is to note whether the user’s overall rating is similar to or different from the positivity of their review content. Finally, as shown in Table 1 each review instance also encodes a vector of the top 50 most popular review topics (*CNT*), indicating whether it is present in the review or not.

$$Density(R_k) = \frac{|pos(topics(R_k))| + |neg(topics(R_k))|}{|topics(R_k)|} \quad (4)$$

### 4.2 Expanding Basic Features

Each of the basic features in Table 1 is calculated for a given single review. For example, we may calculate the *breath* of review  $R_i$  to be 5, indicating that it covers 5 identified topics. Is this a high or low value for the product in question, which may have tens or even hundreds of reviews written about it? For this reason, in addition to this basic feature value, we include 4 other variations as follows to reflect the distribution of its values across a particular product:

- The *mean* value for this feature across the set of reviews for the target product.
- The *standard deviation* of the values for this feature across the target product reviews.

Table 1: Classification Feature Sets.

Type	Feature	#	Description
AGE	<i>Age</i>	1	The number of days since the review was posted.
RAT	<i>NormUserRating</i>	1	A normalised rating score obtained by scaling the user’s rating into the interval $[0, 1]$ .
SIZE	<i>NumSentences</i>	1	The number of sentences in the review text.
	<i>NumWords</i>	1	The total number of words in the review text.
TOP	<i>Breadth</i>	1	The total number of topics mined from the review.
	<i>Depth</i>	1	The average number of words per sentence containing a mined topic.
	<i>Redundancy</i>	1	The total word-count of sentences that are not associated with any mined topic.
	<i>TopicRank</i>	1	The sum of the reciprocal popularity ranks for the mined topics present; popularity ranks are calculated across the target product.
SENT	<i>NumPos (Neg, Neutral)</i>	3	The number of positive, negative, and neutral topics, respectively.
	<i>Density</i>	1	The percentage of review topics associated with non-neutral sentiment.
	<i>NumUPos (Neg, Neutral)</i>	3	The number of <i>unique</i> topics with positive/negative/neutral sentiment.
	<i>WPos (Neg, Neutral)</i>	3	The number of positive, negative, and neutral topics, weighted by their reciprocal popularity rank.
	<i>RelUPos (Neg, Neutral)</i>	3	The relative proportion of unique positive/negative/neutral topics.
	<i>SignedRatingDiff</i>	1	The value of <i>RelUPos</i> minus <i>NormUserRating</i>
	<i>UnsignedRatingDiff</i>	1	The absolute value of <i>RelUPos</i> minus <i>NormUserRating</i>
READ	<i>NumComplex</i>	1	The number of ‘complex’ words (3 or more syllables) in the review text.
	<i>SyllablesPerWord</i>	1	The average number of syllables per word
	<i>WordsPerSen</i>	1	The average number of words per sentence
	<i>GunningFogIndex</i>	1	The number of years of formal education required to understand the review.
	<i>FleschReadingEase</i>	1	A standard readability score on a scale from 1 (30 - very difficult) to 100 (70 - easy).
	<i>KincaidGradeLevel</i>	1	Translates FleschReadingEase into KincaidGradeLevel required (U.S. grade level).
	<i>SMOG</i>	1	Simple Measure of Gobbledygood (SMOG) estimates the years of education required, see [DuBay, 2004].
CNT		50	The top 50 most frequent topics that occur in a particular product’s reviews.

- The *normalised* value for the feature based on the number of standard deviations above (+) or below (-) the mean.
- The *rank* of the feature value, based on a descending ordering of the feature values for the target product.

Accordingly most of the features outlined in Table 1 translate into 5 different actual features (the original plus the above 4 variations) for use during classification. This is the case for every feature (30 in all) in Table 1 except for the content features (*CNT*). Thus each review instance is represented as a total set of 200 features ( $(30 \times 5) + 50$ ).

## 5 Evaluation

We have described techniques for extracting topical features from product reviews and an approach for assigning sentiment to review sentences that cover these topics. Our hypothesis is that these topical and sentiment features will help when it comes to the automatic classification of user generated reviews, into *helpful* and *unhelpful* categories, by improving classification performance above and beyond more traditional features (e.g. terms, ratings, readability); see [Kim *et al.*, 2006; O’Mahony *et al.*, 2009]. In this section we test this hypothesis on real-world review data for a variety of product categories using a number of different classifiers.

### 5.1 Datasets & Setup

The review data for this experiment was extracted from Amazon.com during October 2012; 51,837 reviews from 1,384

unique products. We focused on 4 product categories — *Digital Cameras (DC)*, *GPS Devices*, *Laptops*, *Tablets* — and labeled them as *helpful* or *unhelpful*, depending on whether their helpfulness score was above 0.75 or not, as described in Section 4. For the purpose of this experiment, all reviews included at least 5 helpfulness scores (to provide a reliable ground-truth) and the helpful and unhelpful sets were sampled so as to contain approximately the same number of reviews. Table 2 presents a summary of these data, per product type, including the average helpfulness scores across all reviews, and separately for helpful and unhelpful reviews.

Category	#Reviews	#Prod.	Avg. Helpfulness		
			Help.	Unhelp.	All
DC	3180	113	0.93	0.40	0.66
GPS Devices	2058	151	0.93	0.46	0.69
Laptops	4172	592	0.93	0.40	0.67
Tablets	6652	241	0.92	0.39	0.65

Table 2: Filtered and Balanced Datasets.

Each review was processed to extract the classification features described in Section 4. Here we are particularly interested in understanding the classification performance of different categories of features. In this case we consider 8 different categories, *AGE*, *RAT*, *SIZE*, *TOP*, *SENT-1*, *SENT-2*, *READ*, *CNT*. Note, we have split the sentiment features (*SENT*) into two groups *SENT-1* and *SENT-2*. The latter contains all of the sentiment features from Table 1 whereas

the former excludes the ratings difference features (signed and unsigned) so that we can better gauge the influence of rating information (usually a powerful classification feature in its own right) within the sentiment feature-set. Accordingly we prepared corresponding datasets for each category (Digital Cameras, GPS Devices, Laptops and Tablets) in which the reviews were represented by a single set of features; for example, the *SENT-1* dataset consists of reviews (one set of reviews for each product category) represented according to the *SENT-1* features only.

For the purpose of this evaluation we used three commonly used classifiers: *RF* (*Random Forest*), *JRip* and *NB* (*Naive Bayes*), see [Witten and Frank, 2005]. In each case we evaluated classification performance, in terms of the area under the ROC curve (AUC) using a 10-fold cross validation.

## 5.2 Results

The results are presented in Figures 3(a-d). In Figures 3(a-c) we show the AUC performance for each classification algorithm (RF, JRip, NB) separately; each graph plots the AUC of one algorithm for the 8 different categories of classification features for each of the four different product categories (DC, GPS, Laptop, and Tablet). Figure 3(d) provides a direct comparison of all classification algorithms (RF, JRip, NB); here we use a classifier using all features combined. AUC values in excess of 0.7 can be considered as useful from a classification performance viewpoint [Streiner and Cairney, 2007]. Overall we can see that RF tends to produce better classification performance across the various feature groups and product categories. Classification performance tends to be poorer for the GPS dataset compared to Laptop, Tablet, and DC.

We know from previous research that ratings information proves to be particularly useful when it comes to evaluating review helpfulness [Kim *et al.*, 2006]. It is perhaps no surprise therefore to see that our ratings-based features perform well, often achieving an  $AUC > 0.7$  on their own; for example in Figure 3(a) we see an AUC of approximately 0.75 for the Laptop and Tablet datasets, compared to between 0.65 and 0.69 for GPS and DC, respectively. Other ‘traditional’ feature groups (AGE, SIZE, READ, and CNT) rarely manage to achieve AUC scores  $> 0.7$  across the product categories.

We can see strong performance from the new topic and sentiment feature-sets proposed in this work. The *SENT-2* features consistently and significantly outperform all others, with AUC scores in excess of 0.7 for all three algorithms and across all four product categories; indeed in some cases the *SENT-2* features deliver AUC greater than 0.8 for DC, Laptop and Tablet products; see Figure 3(a)). The *SENT-2* feature group benefits from a combination of sentiment and ratings based features but a similar observation can be made for the sentiment-only features of *SENT-1*, which also achieve AUC greater than 0.7 for almost classification algorithms and product categories. Likewise, the topical features (*TOP*) also deliver a strong performance with  $AUC > 0.7$  for all product categories except for *GPS*.

These results bode well for a practical approach to review helpfulness prediction/classification, with or without ratings data. The additional information contained within the topical

and sentiment features contributes to an uplift in classification performance, particularly with respect to more conventional features that have been traditionally used for review classification. In Figure 3(d) we present summary classification results according to product category when we build classifiers using the combination of all types of features. Once again we can see strong classification performance. We achieve an AUC of more than 0.7 for all conditions and the *RF* classifier delivers an AUC close to 0.8 or beyond for all categories.

## 6 Recommending Helpful Reviews

In many situations users are faced with a classical information overload problem: sifting through potentially hundreds or even thousands of product opinions. Sites like Amazon collect review helpfulness feedback so that they can rank reviews by their average helpfulness scores but this is far from perfect. Many reviews (often a majority) have received very few or no helpfulness scores. This is especially true for more recent reviews, which arguably may be more reliable in the case of certain product categories (e.g. hotel rooms). Moreover, if reviews are sorted by helpfulness then it is unlikely that users will get to see those yet to be rated making it even less likely that they will attract ratings. It quickly becomes a case of “*the rich get richer*” for those early rated helpful reviews. This is one of the strong motivations behind our own work on review classification, but can our classifier be used to recommend helpful reviews to the end user?

Amazon currently adopts a simple approach to review recommendation, by suggesting the most helpful positive and most helpful critical review from a review collection. To evaluate the ability of our classifier to make review recommendations we can use the classification confidence as one simple way to rank-order helpful reviews and select the top-ranked review for recommendation to the user. In this experiment we select the single most confident helpful review for each individual product across the four different product categories; we refer to this strategy as *Pred*. Remember we are making this recommendation without the presence of actual helpfulness scores and rely only on our ability to *predict* whether a review will be helpful. In this experiment we use an RF classifier using all features. As a baseline recommendation strategy we also select a review at random; we call this strategy *Rand*.

We can test the performance of these recommendation techniques in two ways. First, because we know the actual helpfulness scores of all reviews (the ground-truth) we can compare the recommended review to the review which has the actual highest helpfulness score for each product, and average across all products in a given product class. Thus the two line graphs in Figure 4 plot the actual helpfulness of the recommended reviews (for *Pred* and *Rand*) as a percentage of the actual helpfulness of the most helpful review for each product; we call this the *helpfulness ratio* (*HR*). We can see that *Pred* significantly outperforms *Rand* delivering a helpfulness ratio of 0.9 and higher compared to approximately 0.7 for *Rand*. This means that *Pred* is capable of recommending a review that has an average helpfulness score that is 90% that of the actual most helpful review.

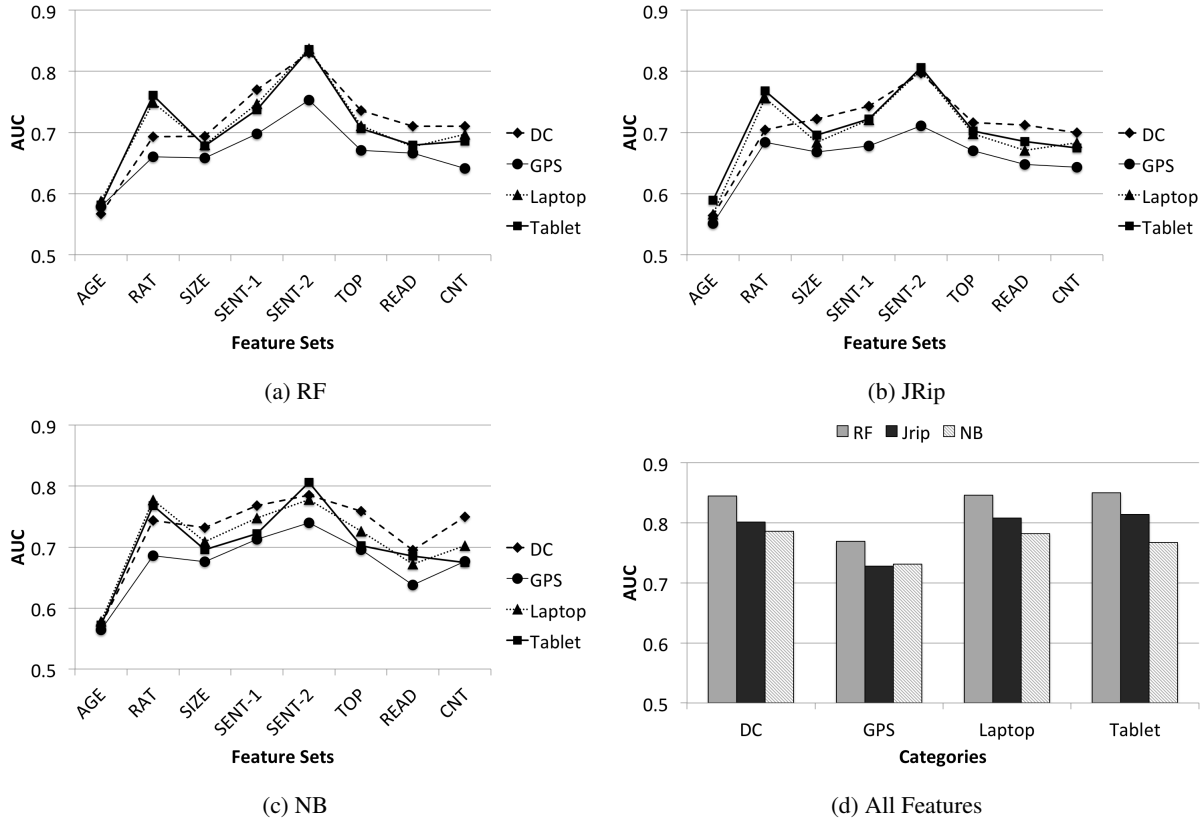


Figure 3: Classification performance results: (a-c) for RF, JRip and NB classifiers and different feature groups; (d) comparison of RF, JRip and NB for all features.

Incidentally, very often the most helpful review has a perfect review score of 1 and this review is often recommended by *Pred*. In fact we can look at this by computing how often *Pred* and *Rand* select their recommended review from among the top  $k$  reviews ranked by actual helpfulness. In Figure 4 we present the results for  $k = 3$  as bars for each product class. For instance, we can see that for Laptops *Pred* recommends the best possible review 60% of the time compared to only about 37% for *Rand*. And in general we can see that *Pred* manages to recommend the optimal review between 1.5 and 2 times as frequently as *Rand*.

In summary we have shown that our helpfulness classifier can be used to recommend helpful reviews, without the need for explicit helpfulness information, and that these recommendations are almost as good as the optimal helpful reviews that could be chosen if perfect helpfulness information was available. Again this bodes well for systems where helpfulness information is not available or is incomplete: it may still be possible to identify and recommend those reviews (new or old) which are likely to be genuinely helpful.

## 7 Conclusion

User-generated reviews are now an important source of knowledge for consumers and are known to play an active role in decision making in many domains. In this paper we have described techniques for mining topical and sen-

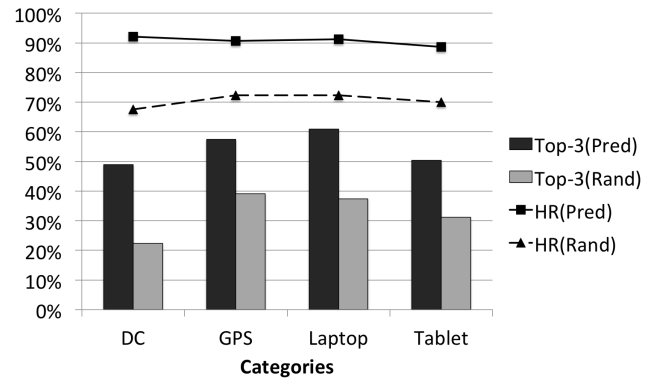


Figure 4: The average helpfulness ratio and top-k results for *Pred* and *Rand* across all product categories.

timent information from user-generated product reviews as the basis for a review quality classification system. We have demonstrated that these topical and sentiment features help to improve classification performance above and beyond that which is possible using more conventional feature extraction techniques. We have further described a possible application of this classification approach and evaluated its ability to make high quality review recommendations in practice.

## References

- [Baccianella *et al.*, 2009] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Multi-facet rating of product reviews. In *Advances in Information Retrieval, 31th European Conference on Information Retrieval Research (ECIR 2009)*, pages 461–472, Toulouse, France, 2009. Springer.
- [Chevalier and Dina Mayzlin, 2006] Judith A. Chevalier and Dina Dina Mayzlin. The effect of word of mouth on sales: Online book reviews. *Journal of Marketing Research*, 43(3):345–354, 2006.
- [Dellarocas *et al.*, 2007] C. Dellarocas, M. Zhang, and N. F. Awad. Exploring the value of online product reviews in forecasting sales: The case of motion pictures. *Journal of Interactive Marketing*, 21(4):23–45, November 2007.
- [Dhar and Chang, 2009] Vasant Dhar and Elaine A. Chang. Does chatter matter? the impact of user-generated content on music sales. *Journal of Interactive Marketing*, 23(4):300–307, 2009.
- [DuBay, 2004] W.H. DuBay. The principles of readability. *Impact Information*, pages 1–76, 2004.
- [Hsu *et al.*, 2009] Chiao-Fang Hsu, Elham Khabiri, and James Caverlee. Ranking comments on the social web. In *Proceedings of the 2009 IEEE International Conference on Social Computing (SocialCom 2009)*, pages 90–97, Vancouver, Canada, 2009.
- [Hu and Liu, 2004a] Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD 2004, pages 168–177, New York, NY, USA, 2004. ACM.
- [Hu and Liu, 2004b] Minqing Hu and Bing Liu. Mining opinion features in customer reviews. *Science*, 4:755–760, 2004.
- [Hu *et al.*, 2008] Nan Hu, Ling Liu, and Jie Zhang. Do online reviews affect product sales? the role of reviewer characteristics and temporal effects. *Information Technology and Management*, 9:201–214, 2008. 10.1007/s10799-008-0041-2.
- [Justeson and Katz, 1995] J. Justeson and S. Katz. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, pages 9–27, 1995.
- [Kim *et al.*, 2006] S.-M. Kim, P. Pantel, T. Chklovski, and M. Pennacchiotti. Automatically assessing review helpfulness. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2006)*, pages 423–430, Sydney, Australia, July 22–23 2006.
- [Li *et al.*, 2011] Fangtao Li, Minlie Huang, Yi Yang, and Xiaoyan Zhu. Learning to identify review spam. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence - Volume Volume Three*, IJCAI 2011, pages 2488–2493. AAAI Press, 2011.
- [Lim *et al.*, 2010] Ee-Peng Lim, Viet-An Nguyen, Nitin Jindal, Bing Liu, and Hady Wirawan Lauw. Detecting product review spammers using rating behaviors. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM 2010, pages 939–948, New York, NY, USA, 2010. ACM.
- [Liu *et al.*, 2008] Y. Liu, X. Huang, A. An, and X. Yu. Modeling and predicting the helpfulness of online reviews. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (ICDM 2008)*, pages 443–452, Pisa, Italy, December 15–19 2008. IEEE Computer Society.
- [Moghaddam and Ester, 2010] Samaneh Moghaddam and Martin Ester. Opinion digger: an unsupervised opinion miner from unstructured product reviews. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM 2010, pages 1825–1828, New York, NY, USA, 2010. ACM.
- [O’Callaghan *et al.*, 2012] Derek O’Callaghan, Martin Horgan, Joe Carthy, and Pádraig Cunningham. Network analysis of recurring youtube spam campaigns. In *ICWSM*, 2012.
- [O’Mahony and Smyth, 2009] M. P. O’Mahony and B. Smyth. Learning to recommend helpful hotel reviews. In *Proceedings of the 3rd ACM Conference on Recommender Systems (RecSys 2009)*, New York, NY, USA, October 22–25 2009.
- [O’Mahony and Smyth, 2010] Michael P. O’Mahony and Barry Smyth. Using readability tests to predict helpful product reviews. In *Adaptivity, Personalization and Fusion of Heterogeneous Information*, RIAO 2010, pages 164–167, Paris, France, 2010.
- [O’Mahony *et al.*, 2009] M. P. O’Mahony, P. Cunningham, and B. Smyth. An assessment of machine learning techniques for review recommendation. In *Proceedings of the 20th Irish Conference on Artificial Intelligence and Cognitive Science (AICS 2009)*, pages 244–253, Dublin, Ireland, 2009.
- [Qiu *et al.*, 2009] Guang Qiu, Bing Liu, Jiajun Bu, and Chun Chen. Expanding domain sentiment lexicon through double propagation. In *Proceedings of the 21st international joint conference on Artificial intelligence*, IJCAI 2009, pages 1199–1204, San Francisco, CA, USA, 2009. Morgan Kaufmann Publishers Inc.
- [Streiner and Cairney, 2007] D.L. Streiner and J. Cairney. What’s under the roc? an introduction to receiver operating characteristics curves. *The Canadian Journal of Psychiatry/La revue canadienne de psychiatrie*, 2007.
- [Witten and Frank, 2005] I.H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.
- [Zhu and Zhang, 2010] Feng Zhu and Xiaoquan (Michael) Zhang. Impact of online consumer reviews on sales: The moderating role of product and consumer characteristics. *Journal of Marketing*, 74(2):133–148, 2010.