Title	Mining Features and Sentiment from Review Experiences
Authors(s)	Dong, Ruihai; Schaal, Markus; O'Mahony, Michael P.; McCarthy, Kevin; Smyth, Barry
Publication date	2013-07-11
Publication information	Delany S.J. and Ontañón S. (eds.). Case-Based Reasoning Research and Development. ICCBR 2013 (Lecture Notes in Computer Science Volume 7969)
Conference details	ICCBR 2013: 21st International Conference on Case-Based Reasoning, Saratoga Springs, New York, USA, 8-11 July 2013
Series	Lecture Notes in Computer Science
Publisher	Springer
Item record/more information	http://hdl.handle.net/10197/8458
Publisher's statement	The final publication is available at Springer via http://dx.doi.org/10.1007/978-3-642-39056-2_5.
Publisher's version (DOI)	10.1007/978-3-642-39056-2_5

Mining Features and Sentiment from Review Experiences

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

CLARITY: Centre for Sensor Web Technologies
School of Computer Science and Informatics
University College Dublin, Ireland
http://www.clarity-centre.org

Abstract. Supplementing product information with user-generated content such as ratings and reviews can help to convert browsers into buyers. As a result this type of content is now front and centre for many major e-commerce sites such as Amazon. We believe that this type of content can provide a rich source of valuable information that is useful for a variety of purposes. In this work we are interested in harnessing past reviews to support the writing of new useful reviews, especially for novice contributors. We describe how automatic topic extraction and sentiment analysis can be used to mine valuable information from user-generated reviews, to make useful suggestions to users at review writing time about features that they may wish to cover in their own reviews. We describe the results of a live-user trial to show how the resulting system is capable of delivering high quality reviews that are comparable to the best that sites like Amazon have to offer in terms of information content and helpfulness.

1 Introduction

User-generated product reviews are now a familiar part of most e-commerce (and related) sites. They are a central feature of sites like Amazon[1], for example, featuring prominently alongside other product information. User-generated reviews are important because they help users to make more informed decisions and ultimately, improve the conversion rate of browsers into buyers [13].

However, familiar issues are starting to emerge in relation to the quantity and quality of user-generated reviews. Many popular products quickly become overloaded with reviews and ratings, not all of which are reliable or of a high quality [6, 9]. As a result some researchers have started to look at ways to measure review quality (by using information such as reviewer reputation, review coverage, readability, etc.) in order to recommend high quality reviews to users [8, 10, 12]. Alternatively, others have focused on supporting users during the review-writing phase [1–3], the intent being to encourage the creation of high quality, more informative reviews from the outset. For example, the work of Healy and Bridge [3]

[1] http://www.amazon.co.uk

proposed an approach to suggest noun phrases, which were extracted from past product reviews that were similar to the review the user was currently writing; see also the work of Dong et al. [1] for a comparison of related approaches. More recently, Dong et al. described a related approach that focused on recommending product topics or features, rather than simple nouns or noun phrases, to users, based on a hand-coded topic ontology [2].

In this work, we focus on supporting the user at the review-writing stage. We describe a browser-based application called the Reviewer's Assistant (RA) that works in concert with Amazon to proactively recommend product features to users that they might wish to write about. These recommendations correspond to product features which are extracted from past review cases; for example, a user reviewing a digital camera might be suggested a feature such as *"image quality"* or *"battery life"*. This paper extends our previous work [2] in two ways. First, unlike our previous work [2], which relied on hand-coded product features/topics, this paper will describe an approach to automatic feature extraction that does not rely on any hand-coded ontological knowledge. Second, in addition to mining topical features we also evaluate the sentiment of these features, as expressed by the reviewer, to capture whether specific product features have been discussed in a positive, negative or controversial sense. For example, a reviewer might be told that *"image quality"* has been previously reviewed positively while *"battery life"* has largely received negative reviews. We demonstrate how these extensions can be added to the RA system and compare different versions, with and without sentiment information, to examine the quality of the reviews produced.

2 Mining Product Review Experiences

This work is informed by our perspective that user-generated product reviews are an important class of *experiential knowledge* and that, by adopting a case-based reasoning perspective, we can better understand the value of these experiences as they are reused and adapted in different ways to good effect. For example, O'Mahony et al. described how past review cases can be used to train a classifier that is capable of predicting review quality [12]. In this paper, we adopt a different challenge. We are interested in supporting the review writing process and we describe how we can do this by reusing similar past review experiences as the basis for recommending topics to a reviewer for consideration.

The summary RA system architecture is presented in Figure 1. Briefly, the starting point for this work is the availability of a case-base of user-generated product review cases $\{R_1, ..., R_n\}$ for a given class of products such as Digital Cameras, for example. These cases are simply composed of the product id, the text of the review, an overall product rating, and a helpfulness score (based on user feedback). The RA system extends these review cases by augmenting them with a set of *review features* $\{F_1, ..., F_m\}$ and corresponding *sentiment scores*, which correspond to the features covered in the review text. These features and scores are automatically mined from the review case-base, mapped back to the

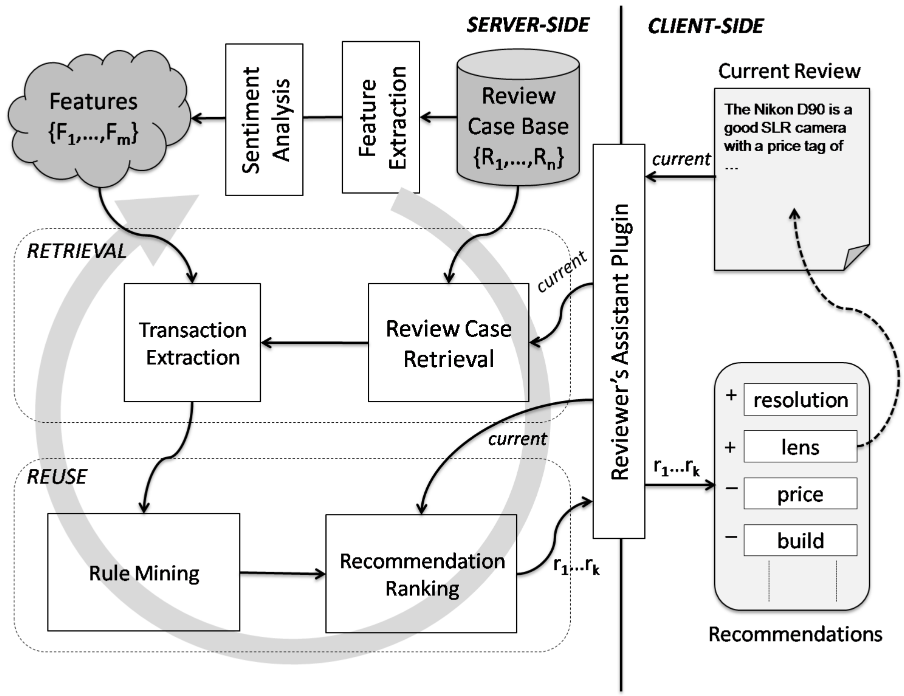


Fig. 1. System architecture.

relevant review text, and then used as the basis for recommendation during review writing as described below.

The client-side component of the RA system is designed as a browser plugin that is 'sensitive' to Amazon's review component, which is to say that it becomes activated when the user lands on a review page. When activated it overlays a set of recommendations $r_1, \ldots, r_k$, marked as the suggestion box in Figure 2. These recommendations are essentially sets of product features that have been automatically mined from past reviews for this product and, by default, they are ranked based on the review text at a particular point in time. In this example, the recommendations are enhanced with additional sentiment information, which has also been mined from past reviews by aggregating the sentiment predictions for different review sentences mentioning the feature in question. The colour of the recommendation indicates the relative sentiment label, whether positive (green), negative (red), controversial (yellow), or without sentiment (blue); controversial features are those which divide reviewer opinions. In addition each feature is annotated with a sentiment bar to visualise the number of positive, negative, and neutral instances for the feature in question. For example, the *battery* feature is marked as negative (red) and the sentiment bar shows that the vast majority of users have reviewed the *battery* of this camera as either negative or neutral, with very few positive opinions expressed.

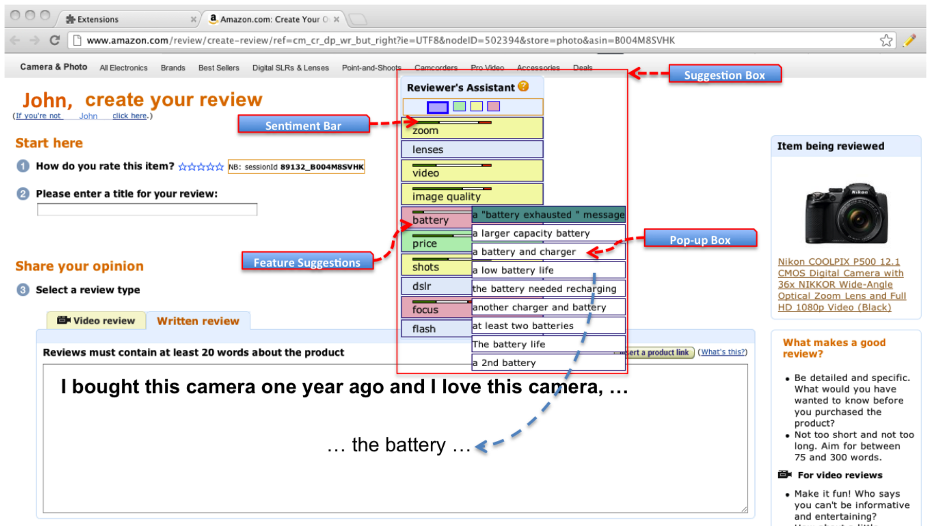


Fig. 2. The RA browser plugin.

2.1 Extracting Review Features

We consider two basic types of review features — *bi-gram* features and *single-noun* features — which are extracted using a combination of shallow NLP and statistical methods, by combining ideas from related research [4, 7]. Briefly, to produce a set of bi-gram features we look for bi-grams in the review cases 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 features but need to be filtered to avoid including AN's that are actually opinionated single-noun features; for example, *great flash* is a single-noun feature (*flash*) and not a bi-gram feature. 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 Hu and Liu's sentiment lexicon [5].

To identify the single-noun topics we extract a candidate set of (non stop-word) nouns from the review cases. Often these single-noun candidates will not make for good case features however; for example, they might include words such as *family* or *day* or *vacation*. The work of Hu and Liu [5] proposes a solution for validating such features by eliminating those that are rarely associated with opinionated words. The intuition is that nouns that frequently occur in reviews and that are often associated with opinion laden words are likely to be popular product features. We calculate how frequently each feature co-occurs with a sentiment word in the same sentence (again, as above, we use Hu and Liu's sentiment lexicon [5]), and retain the single-noun only if its frequency is greater than some threshold (in this case 30%).

This produces a set of bi-gram and single-noun features which we further filter based on their frequency of occurrence in the review cases, keeping only those features ($\{F_1, \ldots, F_m\}$) that occur in at least k reviews out of the total number of n reviews; in this case, for bi-gram features we set $k_{bg} = n/20$ and for single noun topics we set $k_{sn} = 10 \times k_{bg}$ via manual testing. The result is a master list of features for a product case-base and each individual case can then be associated with the set of features that occur within its review text.

2.2 Evaluating Feature Sentiment

Next for each case feature we can evaluate it's sentiment based on the review text that covers the feature. To do this we use a modified version of the *opinion pattern mining* technique proposed by Moghaddam and Ester [11] for extracting opinions from unstructured product reviews. Once again we use the sentiment lexicon from Hu and Liu [5] as the basis for this analysis. For a given feature, F_i, and corresponding review sentence, S_j, from review case C_k (that is the sentence in C_k that mentions F_i), we determine whether there are any sentiment words in S_j. If there are not then this feature is marked as *neutral*, from a sentiment perspective. If there are sentiment words ($w_1, w_2, ...$) then we identify that word (w_{min}) which has the minimum word-distance to F_i.

Next we determine the part-of-speech (POS) tags for w_{min}, F_i and any words that occur between w_{min} and F_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..."* then w_{min} is the word *"great"* which corresponds to an opinion pattern of *JJ-TOPIC* as per [11].

Once an entire pass of all features 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 some minimum number of cases (we use a threshold of 2). 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 either side of of w_{min}. If there are no such negation terms then the sentiment assigned to F_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.

As a result our review cases now include not only the product features identified in their text but also the sentiment associated with these features (positive, neutral, negative). Each of these features is also linked to the relevant fragment of text in the review.

2.3 Reusing Review Cases for Feature Recommendation

For the RA system the primary purpose of review cases is to provide product insights to reviewers for consideration as they write new reviews. This means recommending product features, from relevant past reviews, which fit the context of the current review. This is triggered as the user is writing their review: whenever the user has written a couple of words, or completed a sentence, for example, the recommender returns a new (or updated) set of recommendations.

The recommendations are ranked by default according to a *relevance* metric based on an association rule mining technique which orders features based on their frequency of occurrence in a subset of the *most similar* reviews to the target review so far. This approach is based on the technique described in Dong et al. [2] and is summarised as follows. The relevance ranking process includes the following key steps: (1) review case retrieval; (2) rule mining; (3) transaction extraction; and (4) recommendation generation.

Review Case Retrieval The current review text is used as a textual query against a relevant set of review cases for the same product to retrieve a set of similar reviews. In the current implementation we rely on a simple term-based Jaccard similarity metric to retrieve a set of review cases that are most similar to the query.

Transaction Extraction Each of these review cases is converted into a set of sentence-level transactions and review-level transactions. Briefly, each sentence is converted into the set of features it mentions. If, for example, the review is *"The camera takes good pictures. A flash is needed in poor light."*, then we would have sentence transactions {*camera, pictures*} and {*flash, light*}. And the review level transaction corresponds to the set of features mentioned in the review; if in the above example the review was made up just of these two sentences then the review-level transaction would be {*camera, pictures, flash, light*}.

Rule Mining We apply standard association rule mining techniques across all transactions from the k similar cases to produce a set of feature-based association rules, ranked in descending order of their confidence. For example, we may identify a rule $weight \rightarrow batterylife$ to indicate that when reviews mention camera weight they tend to also discuss battery-life.

Recommendation Ranking To generate a set of ranked recommendations we apply each of the extracted rules, in order of confidence, to the features of the current review text. If the current review text triggers a rule of the form $F_x \rightarrow F_y$, that is because it mentions feature F_x, then the feature F_y is added to the recommendation list. This process terminates when a set of k recommendations has been generated.

2.4 Discussion

This completes our overview of the RA system. Its aim is to provide users with targeted product feature suggestions based on their review to date and the features discussed in similar reviews that have proven to be helpful in the past.

Ultimately our objective with this work is to make a *systems* contribution. That is to say our aim is to develop a novel system and evaluate it in the context of a realistic application setting. Specifically, the primary contribution of this work is to describe the RA as a system that combines automatic feature extraction and sentiment analysis techniques as part of a recommendation system that is designed to support users during the product review process. This builds on previous work by Dong et al. [2] but distinguishes itself in two important ways: (1) by the use of automatic techniques for feature extraction, versus hand-crafted topics; and (2) by exploring the utility of sentiment as part of the recommendation interface.

3 Evaluation

How well does the RA system perform? Does it facilitate the generation of high quality reviews? How do these reviews compare with the best of what a site like Amazon has to offer? What is the impact of including sentiment information as part of the recommendations made to reviewers? These are some of the questions that we will seek to answer in this section via an initial live-user trial of the new RA system.

3.1 Setup

This evaluation is based on an authentic digital camera product review set containing 9,355 user-generated reviews for 116 distinct camera products mined from Amazon.com during October 2012. We implemented two versions of the RA system: (1) RA, which uses automatic feature extraction but does not use sentiment information; (2) $RA+S$, which uses automatic feature extraction and uses sentiment information to distinguish between, for example, positive and negative features as part of the RA recommendation interface.

For the purpose of this evaluation we recruited 33 participants (mainly college students and staff with ages between 17 and 50). These trial participants were mostly novice or infrequent review writers. When asked, 48% (16 out of 33) said they had never submitted an online product review and of those who had, 65% (11 out of 17) of them had written less than 5 product reviews. Each participant was randomly assigned to one of the versions of the RA system; 17 participants were assigned to RA and 16 were assigned to $RA + S$. Each participant was asked to produce a review of a digital camera that was familiar to them and the text of their review was stored for later analysis.

As a competitive baseline for review quality we also extracted 16 high-quality camera reviews from the Amazon data-set; we will refer to these as the $Amazon(+)$ review set. In order to ensure comparability, we chose these reviews of be of similar lengths as the ones created manually with the help of RA and $RA+S$. These 16 reviews were chosen from the subset of the most helpful Amazon reviews by only selecting reviews with a helpfulness score of greater than 0.7. As a result the average helpfulness score of these $Amazon(+)$ reviews was 0.86, meaning that 86% of users found them to be helpful. These are clearly among the best of the user-generated reviews found on Amazon for digital cameras. Therefore this constitutes a genuinely challenging baseline review-set against which to judge the quality of the reviews produced by the trial participants.

3.2 Depth, Breadth & Redundancy

We describe a quantitative analysis of the three sets of reviews (RA, $RA+S$ and $Amazon(+)$) by adopting the approach taken by Dong et al. [2]. For each review we note its length and compute its *breadth*, *depth* and *redundancy*. Briefly, the *breadth* of a review is the number of product features covered by the review. The *depth* of a review is the number of words per feature; that is the word-count of the sentences referring to a given feature. And finally, the *redundancy* of a review is the word-count of the sentences that are not associated with any particular feature.

	$RA+S$	RA	$Amazon(+)$
Breadth*	8.44	7.53	3.63
Depth*	9.41	9.01	17.23
Redundancy**	3.75	10.24	23.63
Length	81.88	81.94	81.50

Table 1. A quantitative analysis of review depth, breadth and redundancy; * indicates pairwise significant difference between Sentiment(RA+S)/ Non-Sentiment(RA) and Amazon+ only, at the 0.05 level; ** indicates significant difference between all pairs at the 0.1 level (using two-tailed t-test).

The result of this analysis, for the three sets of reviews, are presented in Table 1 as averages for review breadth, depth, redundancy and length. We can see that both RA systems (RA and $RA + S$) deliver reviews that are broader (greater feature coverage) than the high-quality Amazon reviews, and with less redundancy. For example, RA and $RA+S$ both lead to reviews that cover more than twice as many product features as the $Amazon(+)$ reviews with less than half of the redundancy. The best performing $RA+S$ condition produces reviews that cover 8.44 product features on average compared to less than 4 product features per review for $Amazon(+)$. Moreover, the $RA+S$ reviews display very low levels of redundancy (3.75 words per review on average) compared to more than 10 and 23 redundant words per review for RA and $Amazon(+)$, respectively. However the reviews produced by RA and $RA + S$ offer less depth of feature coverage than $Amazon(+)$, so although RA and $RA+S$ participants are writing about more features, they are not writing as much about each individual feature.

In relation to the breadth differences, our view is that the RA system helps take some of the "guess work" out of the review-writing process. Reviewers have instant access to a list of meaningful product features (and examples of what other reviewers have written about these features). This reduces some of the friction that is inherent in the review-writing process since the users are no longer solely responsible for prioritising a set of features to write about. Thus users find it easier to identify a set of features to write about and they are naturally inclined to discuss more of these features.

Concerning the difference in depth between the sets of reviews, it is reasonable to take review length as a proxy for the amount of time that users spend writing a review. All three sets of reviews are similar in this regard. Then, per unit time spent writing a review, it is perhaps not surprising that the $Amazon(+)$ reviews enjoy improved depth of feature coverage when compared to RA and $RA+S$; if all 3 sets of users are spending the same time on reviews and $Amazon(+)$ reviewers are covering fewer features, then either they are covering these features in greater depth or they are including more redundant sentences in their reviews. As it turns out both effects are evident: there is a greater depth of coverage for the $Amazon(+)$ reviews but there is also a significant amount of additional redundancy.

There is less of a difference between the RA and $RA+S$ conditions. The additional depth and breadth values for $RA+S$ compared with RA are not statistically significant in this trial. It is worth noting, however, that $RA+S$ does enjoy significantly less levels of redundancy than the RA reviews (an average of 3.75 versus 10.24 redundant words per review). Given that RA and $RA+S$ reviews are similar in terms of depth and breadth, then perhaps there are other metrics that might help us to understand other meaningful differences between these review sets — we consider such metrics in the following sections.

Finally, we appreciate that our measurement of breadth, depth and redundancy depends on the performance of our feature extraction method and so we examined its accuracy against the Amazon data-set. We randomly selected 200 sentences from the more than 99,000 review sentences contained in the 9,355 reviews. From each of these sentences we manually identified a set of features (typically a word or pair of words) and manually judged their sentiment as positive, negative or neutral. This manual annotation process was conducted by 4 independent 'experts' and serves as our ground-truth. We compared our predicted features (sentence by sentence) to the ground-truth for the corresponding sentences and found a precision of 63% and a recall of 67%. The overall accuracy of sentiment prediction is 71%. While these results indicate that there is scope to improve our feature extraction method, it is important to note that the results correspond to a strict matching criterion, i.e. a predicted feature *lens* would not match a ground-truth feature *lens quality*. Given this approach and the large (and statistically significant) differences in breadth, depth and redundancy between the $RA+S/RA$ and $Amazon(+)$ reviews, we believe that the findings as reported above reflect true differences in performance.

3.3 Sentiment Density

Clearly the process by which $RA + S$ reviews are produced is different in one important way from the process that produces RA reviews. The former is informed by indicators of sentiment attached to recommended features. Do these labels influence the actual reviews that are produced? Are users more likely to express opinions on sentiment-laden features?

One way to explore this is to look at what we call the *sentiment density* of a review, by which we mean the percentage of sentences that discuss features in an opinionated manner. The intuition here is that reviews that contain content that is neutral is likely to be less useful, when it comes to making a decision. Sentiment density can be calculated in a straightforward fashion by counting the number of review features with positive or negative sentiment as a fraction of the total number of features in reviews.

	RA+S	*RA*	*Amazon*(+)
Density*/**	65%	48%	49%

Table 2. The sentiment density of RA, $RA + S$ and $Amazon(+)$ reviews; * indicates significant difference between Sentiment and Non-Sentiment at the 0.05 level; ** indicates significant difference between $RA + S$ and Amazon(+) at the 0.1 level (using two-tailed t-test).

Table 2 presents the sentiment density results for our three sets of reviews and clearly points to a significant benefit for those produced using the $RA + S$ condition. The sentiment density of the $RA+S$ reviews is 65% compared to 48% and 49% for the non-sentiment RA and $Amazon(+)$ conditions. In other words, almost two thirds of the features discussed in $RA + S$ reviews are discussed in an opinionated manner; i.e. the reviewer expresses a clear positive or negative viewpoint. By comparison a little less that half of the features mentioned in the RA and $Amazon(+)$ reviews are discussed in an opinionated manner.

As a result, one might expect there to be some benefit in the utility of the $RA + S$ reviews, at least in so far as they contain opinions or viewpoints that are more likely to influence buyers. Clearly the sentiment information that is presented alongside the feature recommendations is influencing users to express stronger (more polarised) opinions for those features that they choose to write about. One caveat here is whether or not the sentiment information is *biasing* what the reviewers write? For example, if they see that *image quality* has been previously reviewed in a positive manner for a particular product, then is the user more likely to write positively about this feature? Obviously this would not be desirable and we will return to this point later.

3.4 Review Quality

Clearly there is a difference between the type of reviews produced with recommendation support (whether with or without sentiment) when compared to the Amazon(+) reviews: both RA and $RA+S$ reviews tend to cover more topics but in less detail than the $Amazon(+)$ reviews; the RA and $RA+S$ reviews contain less redundancy; and the $RA+S$ reviews tend to contain more opinionated content. But how does this translate into the perceived utility of these reviews from a user perspective? The $Amazon(+)$ reviews have been selected from among the most helpful of Amazon's reviews. How will the reviews produced by the less experienced reviewers using RA and $RA+S$ compare?

To answer this question we recruited a set of 12 people to perform a blind evaluation of the three sets of reviews. Each evaluator was asked to rate the *helpfulness, completeness* and *readability* of the reviews on a 5-point scale (with a rating of 1 indicating 'poor' and a rating of 5 indicating 'excellent'). Every review was evaluated by 3 of the 12 participants and their ratings were averaged to calculate mean helpfulness, completeness and readability scores for each set of reviews.

	$RA+S$	RA	$Amazon(+)$
Helpfulness	3.42 (4)	3.33 (3)	3.23 (3)
Completeness	3.06 (3)	3.08 (3)	2.71 (3)
Readability	3.60 (4)	3.51 (4)	3.69 (4)

Table 3. A qualitative analysis of review quality showing mean (median) ratings.

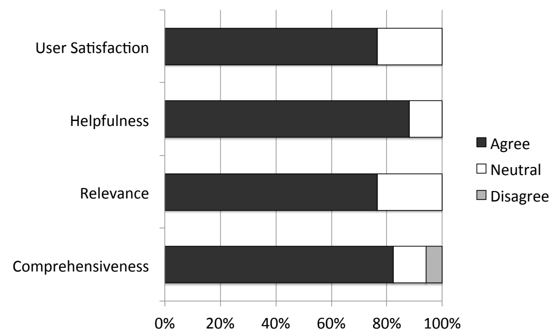
The results are presented in Table 3 as mean and median (bracketed) ratings. As expected the $Amazon(+)$ reviews are rated highly, they are after all among the best reviews that Amazon has to offer. Importantly, we can see however that the reviews produced using the RA and $RA+S$ conditions perform equally well and, in fact, marginally better in terms of review helpfulness and completeness. Although these findings are not definitive — the differences were not found to be statistically significant, not surprising given the scale of the trial — the data bodes well for the approach we are taking. At the very least the additional breadth of coverage offered by RA and $RA+S$ reviews is found to be just as helpful as the best Amazon reviews, for example.

3.5 System Usability & Influence

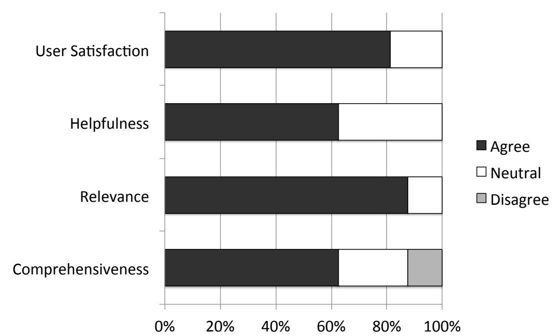
At the end of the trial each participant was also asked to rate the RA system on a 3-point scale (*agree, neutral, disagree*) under the following criteria:

1. *User Statisfaction* – Were you satisfied with the overall user experience?
2. *Helpfulness* – Did the RA help you in writing a review?

3. *Relevance* – Were the specific recommendations relevant to the review you were writing?
4. *Comprehensiveness* – Did the recommendations comprehensively cover the product being reviewed?



(a) RA non-sentiment version.



(b) RA+S sentiment version.

Fig. 3. User feedback.

The results of this feedback for RA and $RA + S$ are presented in Figures 3(a) and 3(b). Broadly speaking users were very satisfied with the RA variations; about 78% of RA users and 82% of $RA + S$ users found the system to be satisfactory and none of the users reported being unhappy with the overall experience. Users also found the reviews to be relevant and mostly helpful, although the $RA + S$ suggestions were judged to be less helpful (62%) that those for the RA system (86%). Interestingly a similar difference is noted with respect to how comprehensive the $RA + S$ suggestions were in comparison to those provided by RA.

Remember that the difference between the RA and $RA + S$ systems is the absence or presence of sentiment information. The above differences would seem to be a result of this interface difference. It is a matter of future work to further explore this by testing different interface choices and different ways to display sentiment information.

Finally,we mentioned earlier the possibility that by displaying sentiment information to users at review time we may lead to biased reviews. As part of the the post-trial feedback (for $RA + S$ participants only) we also asked them to comment on this aspect of the trial as follows:

1. *Influence* – Do you think that the sentiment information influences your own judgement?
2. *Encouragement* – Does the additional sentiment information encourage you to write about your own judgement?
3. *Interruption* – Do you think the additional sentiment information interrupted the review writing process?

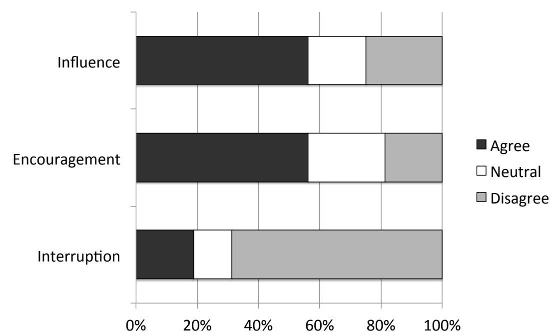


Fig. 4. User feedback on influence, encouragement and interruption – RA+S version.

The results are presented in Figure 4. On the positive side, the participants agreed strongly that the recommendations did not interrupt the review writing process. This finding is not surprising since, as above, participants found the recommendations to be mostly helpful and relevant. A majority of $RA + S$ participants (58%) felt that the availability of sentiment information actually encouraged them to write about features, with less than 20% disagreeing with this proposition. Again this is not surprising given that the $RA + S$ reviews benefit from improved breadth characteristics in particular.

However, a small majority of participants (58%) also felt that the availability of sentiment information was likely to influence the reviews they wrote. This may be an issue and certainly raises the need for additional work to explore this particular aspect of the $RA+S$ system, especially if it turns out to be responsible for reviews that are biased with respect to the sentiment of the recommended features.

3.6 Discussion

The primary objective of this work has been to explore the role of the RA system when it comes to helping users to write high quality reviews based on the recommendation of mined features and sentiment information. The evidence suggests that there are good reasons to be optimistic about this approach. For example, the overall review quality, completeness, and readability of reviews produced using RA and $RA+S$ is at least equivalent to the best of Amazon's reviews even though they were produced by more novice reviewers. The reviews produced with support from RA and $RA+S$ tend to offer broader coverage of product features with less redundancy and so, perhaps, provide a useful counterpoint to the more in-depth Amazon reviews that tend to focus on a narrower set of product features.

There are a number of questions that remain to be answered. For example, there is evidence, as discussed above, that the display of sentiment information at review writing time may exert undue influence over reviewers, which may lead to more biased reviews. It remains to be seen whether this will help users to make more informed decisions than with less opinionated reviews.

Of course there are limitations to the evaluation we have presented in this work. On the positive side it is a genuine attempt to evaluate a working system in a realistic context using independent trial participants and real products. However, it is a small-scale evaluation and although some performance differences were found to be statistically significant, others were not, which ultimately limits what we can conclude from the results. Of course our future work will seek to expand this evaluation to a larger set of users. Nevertheless the results presented do provide compelling evidence that the RA system is providing a useful service. In particular, it is worth re-emphasising that the baseline Amazon reviews chosen as a benchmark were selected among the best quality Amazon reviews available, and so represent a particularly high benchmark for our evaluation.

4 Conclusions

This paper describes an experience-based recommender system that is designed to help users to write better product reviews by passively making suggestions to reviewers as they write. It extends the work of Dong et al. [2] in two important ways. First it is based on a fully automatic approach to review feature extraction without the need for hand-crafted topics or ontologies as in [2]. Secondly, it explores the use of feature sentiment during recommendation and presentation. We have described the results of a detailed live-user trial to consider review quality in terms of metrics, such as feature depth, breadth and sentiment density, demonstrating the quality of RA reviews compared to the best that sites like Amazon has to offer.

5 Acknowledgments

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

References

1. Ruihai Dong, Kevin McCarthy, Michael P. O'Mahony, Markus Schaal, and Barry Smyth. Towards an intelligent reviewer's assistant: Recommending topics to help users to write better product reviews. In *Procs. of IUI: 17th International Conference on Intelligent User Interfaces, Lisbon, Portugal, February 14-17, 2012*, pages 159–168, 2012.
2. Ruihai Dong, Markus Schaal, Michael P. O'Mahony, Kevin McCarthy, and Barry Smyth. Harnessing the Experience Web to Support User-Generated Product Reviews. In *20th International Conference on Case-Based Reasoning, Lyon, France*, 2012.
3. Paul Healy and Derek Bridge. The GhostWriter-2.0 System: Creating a Virtuous Circle in Web 2.0 Product Reviewing. In Derek Bridge, Sarah Jane Delany, Enric Plaza, Barry Smyth, and Nirmalie Wiratunga, editors, *Procs. of WebCBR: The Workshop on Reasoning from Experiences on the Web (Workshop Programme of the Eighteenth International Conference on Case-Based Reasoning)*, pages 121–130, 2010.
4. 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 '04, pages 168–177, New York, NY, USA, 2004. ACM.
5. Minqing Hu and Bing Liu. Mining opinion features in customer reviews. *AAAI' 04*, 4:755–760, 2004.
6. Nitin Jindal and Bing Liu. Review spam detection. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 1189–1190, New York, NY, USA, 2007. ACM.
7. J. Justeson and S. Katz. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, pages 9–27, 1995.
8. 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.
9. Theodoros Lappas. Fake reviews: The malicious perspective. In Gosse Bouma, Ashwin Ittoo, Elisabeth Mtais, and Hans Wortmann, editors, *Natural Language Processing and Information Systems*, volume 7337 of *Lecture Notes in Computer Science*, pages 23–34. Springer Berlin / Heidelberg, 2012.
10. 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.
11. 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 '10, pages 1825–1828, New York, NY, USA, 2010. ACM.
12. 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.
13. 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.