



<b>Title</b>	The Reviewer's Assistant: Recommending Topics to Writers by Association Rule Mining and Case-base Reasoning
<b>Authors(s)</b>	Dong, Ruihai, Schaal, Markus, O'Mahony, Michael P., Smyth, Barry
<b>Publication date</b>	2012-09-17
<b>Publication information</b>	Dong, Ruihai, Markus Schaal, Michael P. O'Mahony, and Barry Smyth. "The Reviewer's Assistant: Recommending Topics to Writers by Association Rule Mining and Case-Base Reasoning," September 17, 2012.
<b>Conference details</b>	The 23rd Irish Conference on Artificial Intelligence and Cognitive Science (AICS 2012), Dublin City University, Dublin, Ireland, 17-19 September, 2012
<b>Item record/more information</b>	<a href="http://hdl.handle.net/10197/4348">http://hdl.handle.net/10197/4348</a>

Downloaded 2026-06-16 01:59:44

The UCD community has made this article openly available. Please share how this access benefits you. Your story matters! (@ucd\_oa)



© Some rights reserved. For more information

# THE REVIEWER'S ASSISTANT: RECOMMENDING TOPICS TO WRITERS BY ASSOCIATION RULE MINING AND CASE-BASE REASONING

*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  
firstname.lastname@ucd.ie  
<http://www.clarity-centre.org>

## ABSTRACT

Today, online reviews for products and services have become an important class of user-generated content and they play a valuable role for countless online businesses by helping to convert casual browsers into informed and satisfied buyers. As users gravitate towards sites that offer insightful and objective reviews, the ability to source helpful reviews from a community of users is increasingly important. In this extended abstract we describe the Reviewer's Assistant, a case-based reasoning inspired recommender system designed to help people to write more helpful reviews on sites such as Amazon and TripAdvisor. In particular, we describe two approaches to helping users during the review writing process and evaluate each as part of a blind live-user study. Our results point to high levels of user satisfaction and improved review quality compared to a control-set of Amazon reviews.

## 1. INTRODUCTION

Customer reviews have become an important part of our online (and indeed real-world) shopping experiences. Today the vast majority of e-commerce sites feature customer reviews prominently in their core product and service listings. Indeed this type of user-generated content has become a vital part of the value proposition of services such as Amazon and TripAdvisor. so much so that users will often use these resources and similar as a source of product reviews even if they do not wish to make a purchase on the site in question. Online reviews can be helpful and can influence the buying patterns of shoppers. For example, [2] examine the influence of online reviews on video game sales, indicating that reviews are more influential for less popular games. Given that people are increasingly turning to user-generated reviews to support their decision-making it is important for sites like Amazon to help people find and create high-quality reviews. Amazon provides users with the opportunity to rate reviews based on their

*helpfulness* and allows prospective customers to rank reviews by their helpfulness score. Moreover, the work of O'Mahony and Smyth [3] explored the potential to automatically determine the helpfulness of reviews, in the face of sparse ratings data.

While many sites prioritise the promotion of user generated review content, they do little to support users when it comes to generating reviews, beyond the provision of a simple text-input review-form. At best, this provides friction when it comes to attracting new reviews from first-time reviewers, but further it may negatively impact the quality of the reviews that are provided. For this reason researchers have begun to consider ways in which users can be better supported during the review writing process. For example, the seminal work of Bridge et al. [4] describes a system called GhostWriter designed with this task in mind. Briefly, GhostWriter uses case-based reasoning techniques to harness a collection of past review experiences, which are then used as a source of suggestions for reviewers as they write. Essentially, GhostWriter suggests fragments of similar reviews as hints to the reviewer.

In this work, we approach the problem of review quality from a different but complementary perspective: rather than attempt to rank reviews by helpfulness, or eliminate biased reviews, we focus on the process of writing a review in the first place. Our aim is not only to produce a better quality of user-generated reviews, but also to increase the number and diversity of reviews by attracting first-time reviewers who might be initially daunted at the prospect of writing a product review. Our work is inspired by the GhostWriter system [4]. We have developed the Reviewer's Assistant (RA) [1], which harnesses past review experiences to suggest topics as the user writes their review. In this paper, we will describe how the RA combines association rule mining and topic extraction techniques with conversational CBR to generate these recommendations. In addition, the RA has been designed to fully integrate with existing online services such as Amazon and TripAdvisor, allowing users to use existing review tools while benefitting from suggestions made by the system.

---

This work is supported by Science Foundation Ireland under grant 07/CE/I1147 and originally published in [1].

## 2. THE REVIEWER'S ASSISTANT

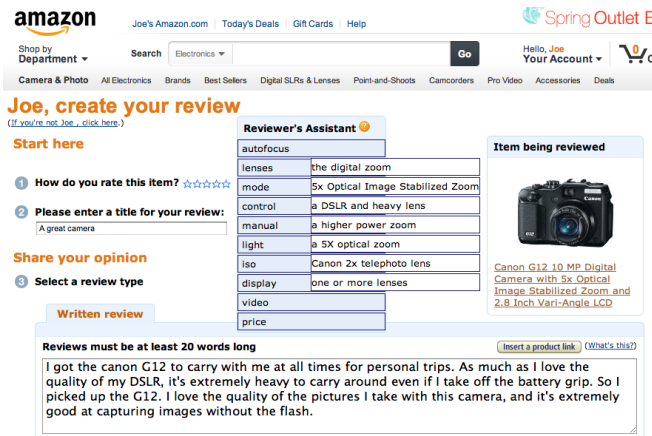


Fig. 1: The Reviewer's Assistant.

The Reviewer's Assistant has been implemented as a browser plugin so that it can seamlessly integrate with pre-existing services like Amazon and TripAdvisor at the user interface level, providing support to users as they write their reviews, but without the need for backend integration with these underlying services. Briefly, the Reviewer's Assistant takes the form of an additional recommendation module that appears on review-creation pages. Figure 1 shows this in the context of an Amazon review page with the RA appearing as a floating set of recommendations which refresh as the user progresses with their review. In this section we will describe the RA system and key technical features.

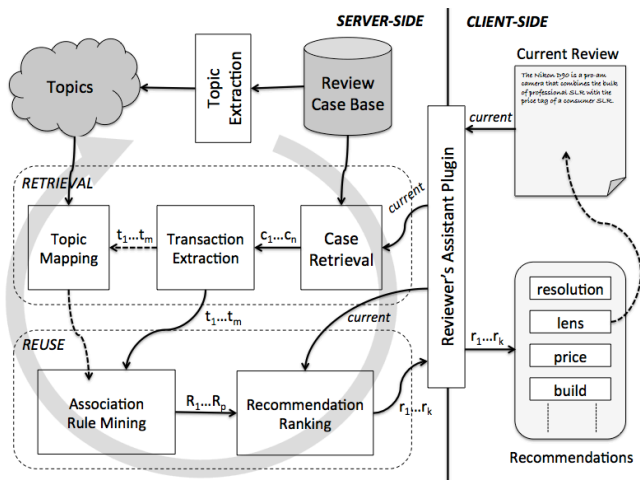


Fig. 2: The Reviewer's Assistant System Architecture.

Figure 2 provides a high-level overview of the Reviewer's Assistant system. The basic recommendation cycle is triggered on an ongoing basis as the user writes their review. During each recommendation cycle the user's current review text

is used to identify a set of similar review cases. These cases are mined to extract a set of term-based association rules. These rules are then used to map the current review text to a ranked list of concrete topic recommendations for the user.

### 2.1. Case Discovery and Topic Extraction

Before describing the technical details of the basic RA recommendation cycle we will first look at the structure and source of case and domain knowledge.

The RA is designed to operate over specific product domains and maintains a separate review case-base for each. A domain is assumed to be made up of a collection of products that share similar features, which ultimately will act as possible review targets for the reviewer. Each case corresponds to a previous review and includes the product id, the review text, and any meta information available, such as the overall review score or helpfulness. These reviews are automatically extracted from the underlying service by using any available API to extract relevant product and review data; typically the RA will extract only high-quality reviews (based on any helpfulness/quality meta data that is available).

Here we consider a version of the RA that maps review terms to product topics in order to improve recommendation quality. For the purpose of our feasibility study we adopted a very simple approach to topic modeling based around a hand-coded set of topics for the target domain, with each topic associated with a synonym set. As such the topic extraction component of Figure 2 was not automated for the variation of the Reviewer's Assistant considered here. However an automated version is currently under development.

### 2.2. Retrieving Similar Reviews

As the user writes their review, the review text is periodically used (typically on the completion of a new sentence) as query *current* against the relevant domain case-base to retrieve a set of similar reviews, from which term-based transactions are extracted as the basis for association rule mining.

In the current implementation we rely on a simple term-based Jaccard similarity metric to retrieve a set of the  $n$  review cases that are most similar to *current*. At present, this retrieval process is further restricted to only consider review cases that match the target review product id.

At this point each of the  $n$  retrieved reviews is converted into a set of sentence-level transactions and a review-level transaction. This is a straightforward process that starts by identifying the nouns in a review text and then converts each sentence or review into a set of these nouns. 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 a review transaction  $\{camera, pictures, flash, light\}$ .

In one version of the RA, *non-topic*, recommendation proceeds based on the mining of these noun-based transactional

representations. However, we also consider a topic-based approach (we refer to this simply as *topic*) which first maps the raw transaction terms onto the topics extracted from a given review collection as mentioned above. The purpose of this approach is that it affords a level of abstraction (topics vs. nouns) that has the potential to provide a more intuitive set of recommendations based on more meaningful product topics, rather than on the looser vocabulary of user generated reviews. As part of our evaluation in Section 3 we will consider whether this topic variation in fact translates into any meaningful evaluation benefit.

### 2.3. Generating Recommendations

The RA generates a set of ranked recommendations by using association rule mining techniques to discover patterns of nouns/topics that recur frequently across many reviews.

Here we have a set of transactions (whether *non-topic* or *topic*), which reflect frequent collections of nouns/topics that occur at the sentence-level or review-level. For example, in the digital camera domain we might have transactions such as  $\{image, lens, resolution\}$  and  $\{size, price\}$  extracted at the sentence-level to indicate that review sentences discussed camera resolution, lens type and image quality or camera size and price. We can apply association rule mining [5] to identify frequently occurring transactions and to generate a set of association rules of the form  $\{image, lens\} \rightarrow \{resolution\}$ .

The resulting rules are ranked in descending order of their confidence, which is basically an estimate of the probability of finding the topic/noun that forms the rule consequent given the occurrence of the antecedent. To generate a set of ranked recommendations we apply each of the extracted rules, in order of confidence, to the current review text. If the current review text triggers a rule of the form  $LHS \rightarrow RHS$  then the noun/topic that is the  $RHS$  is added to the recommendation list. This process terminates when a set of  $k$  recommendations have been generated.

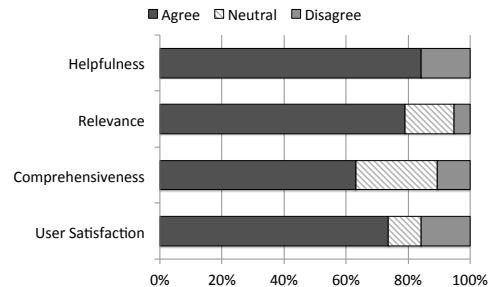
## 3. EVALUATION

In this section we describe two experiments to evaluate the utility of the Reviewer’s Assistant in practice. We pay particular attention to performance differences between the *non-topic* and *topic* variations of the RA, if any. First we describe a live-user study to understand the end-user experience of using RA. Second we use the reviews produced during this first experiment as input to a follow-up study of review quality, comparing these reviews to reviews on Amazon.

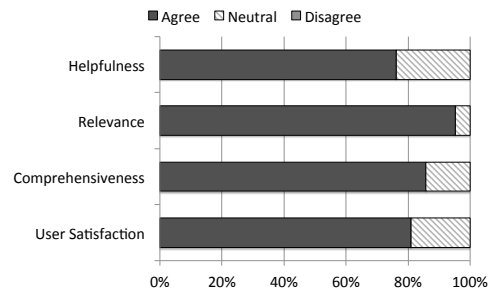
### 3.1. Usage Analysis

For the first part of this experiment we recruited 40 test users. We restricted our target product domain to that of digital cameras on Amazon and configured the RA plugins (*non-topic*

and *topic*) accordingly; we chose this product domain because all users had at least some experience with this type of product. The participants were randomly divided into *non-topic* and *topic* groups; exactly 21 participants (52.5%) had access to the *non-topic* version of the RA whilst the remaining 19 (47.5%) used the *topic* version. Each user was asked to select a product of interest and to write a review for this product; they were provided with a brief initial tutorial on the RA, the purpose of its suggestions, and how they might avail of them if appropriate.



(a) Topic suggestions.



(b) Non-topic suggestions

**Fig. 3: User Feedback.**

The results of this questionnaire are shown in Figure 3 and are largely positive with respect to both the *topic* and *non-topic* variations. For example, we can see that overall about 75% – 85% of users found the RA to be helpful, with the higher percentage pertaining to the *topic* variation. Interestingly, this advantage for *topic* is reversed when we look how relevant users found the recommendations to be. In this case we can see that, while 90% of the *non-topic* users rated the recommendations to be relevant, only 80% of the *topic* users rated their recommendations as relevant. Similarly, the feedback in terms of recommendation comprehensiveness also favours the *non-topic* variation, with scores of 85% versus 65% for *non-topic* and *topic* users, respectively.

Finally, in relation to the post-trial questionnaire we can see that overall there is strong user-support for the RA. Between 75% (*topic*) and 80% (*non-topic*) of users indicated that they were satisfied overall with the system.

### 3.2. Review Quality

Ultimately the best test of the RA approach is to consider the quality of the resulting reviews in order to understand whether users find them to be helpful, for example. Even better is if we can compare our test reviews to a benchmark in terms of quality. This is the aim of this final evaluation section.

We collected 2 sets of reviews with similar lengths. The first set were chosen at random from the reviews written by participants of the RA trial above. We collected 10 random reviews written using the help of the RA with *topic* and another 10 written using the help of the RA with *non-topic*. For our second set, we selected two groups of Amazon reviews to serve as a benchmark, against which to judge the quality of the RA reviews. One group was chosen at random from among the most helpful Amazon camera reviews. We picked 10 reviews (*Amazon+*) that had a helpfulness score of at least 0.7 (meaning 70% of raters considered them helpful); in fact, the average helpfulness score for these reviews was 0.9 and thus we can view these as examples of very high quality product reviews written without the aid of RA. Next we chose another group of 10 random Amazon reviews (*Amazon-*), but this time we picked reviews that had a helpfulness score of less than 0.7; the average helpfulness score for reviews in this group was 0.41 and thus represent examples of lower quality reviews written without the help of RA.

	topic	non-topic	Amazon+	Amazon-
Helpfulness*	3.90	3.90	3.33	3.07
Completeness*	3.67	3.57	2.67	2.53
Readability	3.60	3.60	3.80	3.33

**Table 1:** User Evaluation; \* indicates significant difference between topic/ non-topic and Amazon+/ Amazon-.

Next, we recruited 15 reviewer “experts” (with a good understanding of the digital camera space) and asked them to perform a blind review of a random sample of reviews from the four sets above (*topic*, *non-topic*, *Amazon+*, and *Amazon-*). In each case we asked the experts to rate the reviews on a 5-point scale in terms of 1) helpfulness – how helpful did they think the review would be to others? 2) completeness – did the review provide a reasonably complete account of the product in question? 3) readability – was the review well written and readable? In total each test review was reviewed by 3 different experts. Finally, we calculated the average helpfulness, completeness, and readability ratings across each of the 4 review groups and also calculated their average breadth, depth and redundancy scores based on the approach taken previously.

The results are presented in Table 1 and show a significant positive benefit accruing to the RA in a number of important respects. For instance the average helpfulness rating of RA reviews (3.90 for both RA versions) is greater than the helpfulness rating for *Amazon+* (3.33) and *Amazon-* (3.07). Sim-

ilarly, we can see clear benefits for the RA variations in terms of review completeness (3.67 and 3.57), when compared to *Amazon+* (2.67) and *Amazon-* (2.53). Both of these helpfulness and completeness benefits (RA versus Amazon) are statistically significant at the 0.05 level; statistically significant differences were not found in terms of review readability.

## 4. CONCLUSIONS

The work presented here [1] was inspired by early work on the GhostWriter systems [4], which highlighted the potential for web experiences and case-based reasoning to support users when creating user-generated content, whether in the form of adverts or product reviews. The main contribution of the work presented in this paper is twofold. Firstly, we have extended the original Ghostwriter approach by incorporating a combination of association rule mining and topic extraction to generate review recommendations that are more likely to match key product features as important review targets. Secondly, we have presented a comprehensive evaluation of the RA system, focusing on the overall user experience and a benchmarked study of real review quality. The results show that users find the RA to be useful and the resulting reviews are rated more highly than comparable Amazon reviews, even when compared against a set of best quality reviews.

## 5. REFERENCES

- [1] 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*, to appear.
- [2] Feng Zhu and Xiaoquan (Michael) Zhang, “Impact of Online Consumer Reviews on Sales: The Moderating Role of Product and Consumer Characteristics,” *Journal of Marketing*, vol. 74, no. 2, pp. 133–148, 2010.
- [3] Michael P. O’Mahony and Barry Smyth, “Learning to Recommend Helpful Hotel Reviews,” in *Proceedings of the third ACM conference on Recommender Systems*. 2009, RecSys ’09, pp. 305–308, ACM.
- [4] Derek Bridge and Paul Healy, “The GhostWriter-2.0 Case-Based Reasoning system for making content suggestions to the authors of product reviews,” *Knowledge-Based Systems*, vol. 29, no. 0, pp. 93 – 103, 2012.
- [5] R Agrawal, Tomasz Imieliński, and A Swami, “Mining Association Rules between Sets of Items in Large Databases,” *ACM SIGMOD Record*, vol. 22, no. May, pp. 207–216, 1993.