Title	Learning to recommend helpful hotel reviews
Authors(s)	O'Mahony, Michael P.; Smyth, Barry
Publication date	2009-10
Publication information	Proceedings of the third ACM conference on Recommender systems
Conference details	Paper presented at the 3rd ACM Conference on Recommender Systems (RecSys 2009), New York City, NY, USA, 22-25 October 2009
Publisher	ACM
Link to online version	http://doi.acm.org/10.1145/1639714.1639774
Item record/more information	http://hdl.handle.net/10197/1894
Publisher's version (DOI)	10.1145/1639714.1639774

Learning to Recommend Helpful Hotel Reviews*

Michael P. O'Mahony
CLARITY: Centre for Sensor Web Technologies
School of Computer Science and Informatics
University College Dublin, Ireland
michael.p.omahony@ucd.ie

Barry Smyth
CLARITY: Centre for Sensor Web Technologies
School of Computer Science and Informatics
University College Dublin, Ireland
barry.smyth@ucd.ie

ABSTRACT

User-generated reviews are a common and valuable source of product information, yet little attention has been paid as to how best to present them to end-users. In this paper, we describe a classification-based recommender system that is designed to recommend the most *helpful* reviews for a given product. We present a large-scale evaluation of our approach using TripAdvisor hotel reviews, and we show that our approach is capable of suggesting superior reviews compared to a number of alternative recommendation benchmarks.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Algorithms, Experimentation

Keywords

Review Recommendation, Classification, TripAdvisor

1. INTRODUCTION

Product reviews are an increasingly important type of user-generated content as they provide a valuable source of information to help customers make good purchasing decisions. Typically, these reviews consist of an overall product score (often in the form of a star-rating) and some free-form review text to allow the reviewer to describe their experience with the product or service in question. In the world of recommender systems, these reviews serve as a type of *recommendation explanation* [2, 5, 9] and help the user to better evaluate the quality of product suggestions.

Insightful product reviews can be extremely helpful in guiding purchasing decisions. Not all reviews, however, are helpful; biased reviews, for example, can be misleading while poorly authored reviews can be difficult to understand. As reviews accumulate, it can become increasingly difficult for users to identify those that are helpful; it is not uncommon for popular products to attract tens, if not hundreds, of reviews, thereby introducing yet another information overload problem. While some services are addressing this by allowing users to rate the helpfulness of each review, this type of feedback can be sparse, with many reviews, particularly more recent ones, failing to attract any feedback.

The increasing availability of product reviews presents a new and challenging recommendation opportunity – to recommend or re-rank reviews according to their likely *helpfulness* – which complements the more traditional product recommendation scenarios. Thus the job of the *product recommender* is to suggest a shortlist of relevant products to the user, while the role of the *review recommender* is to suggest helpful reviews for each of these products.

In this paper, we describe and evaluate a review recommender system. Related work is described in Section 2. In Section 3, we adopt a classification approach to harness available review feedback to learn a classifier that is capable of identifying helpful and non-helpful reviews. We then describe how this classifier can be used as the basis for a practical recommendation technique that seeks to automatically suggest the most-helpful contrasting reviews to the end-user. An evaluation of our approach, based on a large set of TripAdvisor hotel reviews, is presented in Section 4.

2. RELATED WORK

In this section, we present a brief review of related work. In [6], SVM regression was used on structural, lexical, syntactic, semantic and sentiment features to rank Amazon product reviews according to their helpfulness. In this work, it was found that the most discriminating features included the length of a review, its unigrams and score. In [7], timeliness was shown to be a good predictor of the helpfulness of movie reviews, where review helpfulness was seen to decline for older reviews. Reviewer expertise was also found to be a useful feature, indicating that reviewers who were familiar with particular movie genres were likely to produce good reviews for movies in the same or similar genres.

In [4], a classification approach was proposed to distinguish between conversational and informational questions in social Q&A sites. Features relating to question category, text categorization and social network metrics were used to train classifiers and good performance was achieved. The effect of credibility on the retrieval of topical blog posts was examined in [10]. Various credibility indicators were consid-

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

RecSys'09, October 23–25, 2009, New York, New York, USA.

ered, including the regularity at which bloggers post, timeliness of posts, post length, spelling quality and the appropriate use of capitalisation and emoticons in the text. These indicators were found to improve retrieval performance [10].

In this paper, we expand on this work by considering feature sets in relation to reviewer reputation, social properties of the user–hotel review graph and features that capture review completeness (see Section 3.1). In addition, we analyse recommendation of the most helpful reviews in relation to sentiment, and we highlight an interesting performance asymmetry that is biased in favour of reviews expressing negative sentiment.

3. CLASSIFYING AND RECOMMENDING REVIEWS

Using the TripAdvisor domain as a case study, in this section we describe our supervised classification approach to identify the most helpful hotel reviews. Importantly, TripAdvisor allows users to provide feedback on whether they found reviews to be helpful or not. We define review *helpfulness* as the percentage of positive *opinions* that a review has received. Review instances are labeled *helpful* or *non-helpful* and are considered *helpful* if and only if $\geq 75\%$ of opinions are positive. In this way the classification task is focused on the prediction of the most (unambiguously) helpful reviews.

Relying on review feedback alone, however, to recommend reviews is insufficient, given that many reviews fail to attract the critical mass of opinions that would permit reliable helpfulness assessments to be made. Our approach seeks to train a classifier from reviews that have attracted a critical mass of helpfulness opinions, such that the classifier can then be used to classify the helpfulness of arbitrary reviews, including those that have not received any helpfulness feedback.

3.1 Classification Features

Prior to classification, reviews are translated into a feature-based instance representation. Review instances consist of features from four feature categories, which are mined from individual reviews and from the wider community reviewing activity. We now describe each feature category in turn.

Reputation features are designed to capture a user's reputation with respect to the set of reviews that the user has authored in the past. The features are: the mean (**R1**) and standard deviation (**R2**) of review helpfulness over all reviews authored by the user; the percentage of reviews authored by the user which have received a minimum of T opinions (**R3**). In this work, $T = 5$ (see Section 4).

Content features are derived from the review text. We consider three such features: the number of terms in the review text (**C1**); the ratio of uppercase and lowercase characters to other characters in the review text (**C2**); the ratio of uppercase to lowercase characters in the review text (**C3**).

We also consider features in relation to review *completeness*, i.e. how much *optional* review content is provided. The following content can be optionally provided in TripAdvisor reviews. Users can specify what they *liked* and *disliked* most about the hotel, and can provide *sub-scores* in relation to certain aspects of the hotel (e.g. *value*, *rooms*, *location* etc.). Further, users can provide some personal information and details relating to the date and purpose of their visit. Finally, users can respond to a set of review-template questions such as *Would I recommend this hotel to my best friend?* and *I recommend this hotel for* etc.

We use the following features in respect of review completeness: an integer which captures whether the user has completed one, both or none of the optional *liked* and *disliked* parts of the review (**C4**); the number of optional personal and purpose of visit details that are provided by the user (**C5**); the number of optional review-template questions that are answered in the review (**C6**).

Social features are derived from the degree distribution of the user–hotel review graph. We mine six such features: the number of reviews authored by the user (**SL1**); the mean (**SL2**) and standard deviation (**SL3**) of the number of reviews authored by all users; the number of reviews submitted for the hotel (**SL4**); the mean (**SL5**) and standard deviation (**SL6**) of the number of reviews submitted for all hotels.

Sentiment features relate to how well users enjoyed their experience with a hotel. In this paper, we consider sentiment in terms of the score and the optional sub-scores (expressed on a 5-star scale) that a user has assigned to a hotel[1]. We extract the following set of features from reviews: the score assigned by the user to the hotel (**ST1**); the number of (optional) sub-scores assigned by the user (**ST2**); the mean (**ST3**) and standard deviation (**ST4**) of the sub-scores assigned by the user; the mean (**ST5**) and standard deviation (**ST6**) of the scores assigned by the user over all reviews authored by the user; the mean (**ST7**) and standard deviation (**ST8**) of the scores assigned by all users to the hotel.

3.2 Recommendation via Classification

Using the collection of review instances as supervised training data, unseen instances (reviews) in the absence of helpfulness data can be classified. In addition, the classifiers we employ can be configured to return a *confidence* score for class predictions. Prediction confidence can then be used to effectively translate review classification into review recommendation, by rank-ordering those reviews classified as helpful according to prediction confidence. In this way, given a set of reviews for a hotel, we can use a review classifier to produce a ranked list of reviews predicted to be helpful.

Of course other recommendation styles are also possible. For example, one approach is to recommend the most-helpful highly-scored and poorly-scored reviews for a hotel to provide the user with contrasting reviews. Amazon has recently started using this style of review recommendation, but it is of course limited to those reviews that have attracted feedback on review helpfulness. The benefit of the approach described in this paper is that it can be used to generate review recommendations for reviews that have not yet attracted any (or a critical mass of) such feedback.

4. EVALUATION

We created two large datasets by extracting all TripAdvisor reviews prior to April 2009 for users who had reviewed at least one hotel in either of two popular US cities, Chicago or Las Vegas. To provide support when labeling review instances, we selected only those reviews which had received a

[1]Sentiment can also be mined from the review text [1, 8], but we do not consider such an approach in this paper.

minimum of $T = 5$ (either positive of negative) opinions as training data. In addition, we sampled from these reviews to produce balanced training data with a roughly equal representation of both *helpful* and *non-helpful* class instances. Here, we report findings for the Las Vegas dataset only; similar results applied for the Chicago dataset. In total, there were 35,802 reviews by 18,849 distinct users on 10,782 hotels in the balanced Las Vegas dataset.

Using Weka's default settings [11], we compared the performance of three classifiers: JRip, J48 and naïve Bayes (NB). JRip provided the best overall performance; thus we report AUC (area under ROC curve) using 10 fold cross-validation for this classifier. AUC produces a value between 0 and 1, with higher values indicating better performance [3].

4.1 Classification Results

In the following sections, the performance obtained across different groupings of features and feature types is described.

4.1.1 Classification using All Features

Let us begin by looking at classification performance when all features (that is, reputation, social, sentiment, content plus three generic features: *user-id*, *hotel-id* and *review date*) are used for classification. AUC results are presented in Figure 1(a), as the bar labeled 'A'. We can see that the JRip classifier performed well, achieving an AUC score of 0.82.

Reputation features are likely to weigh heavily with respect to classification performance. After all, these features include information about how helpful the review author has proven to be in authoring other reviews. For this reason we have also included results for training instances that include all features except reputation features, condition 'A\R' in Figure 1(a). As expected, we see a drop in classification performance suggesting that the reputation features do in fact play an important role. We will return to this point in the next section, but for now we note that even in the absence of reputation features (and remember that these features will not be available in all domains) classification performance remained high, with an AUC score of 0.74 being achieved.

4.1.2 Classification by Feature Category

The performance of classifiers trained using the reputation, social, sentiment, and content feature categories are also presented in Figure 1(a), as bars labeled 'R', 'SL', 'ST' and 'C', respectively. In particular, the results highlight the strong performance of the reputation features. Overall, reputation features provided best performance, followed by sentiment features, with AUC scores of 0.78 and 0.71 being achieved for these feature categories, respectively. Social and content features were less successful; in both cases, an AUC score of 0.59 was achieved for these categories.

4.1.3 Feature Selection

The analysis presented above examined the relative importance of the different feature categories. Such an analysis does not, however, consider the relative importance of individual features. Thus we show in Table 1 the top 9 features, which are rank-ordered according to *information gain* (IG).

As expected, the reputation features proved to very significant; for example, the mean helpfulness of a user's reviews (**R1**) turned out to be the strongest single predictor of classification accuracy. A total of 4 sentiment features (**ST1**, **ST3**, **ST5** and **ST6**) were ranked among the top features, reflecting the relatively high classification performance achieved by such features as shown in Figure 1(a).

Table 1: Features ranked by information gain (IG).

Rank	Feature ID	IG
1	R1	0.172
2	ST1	0.095
3	ST3	0.079
4	ST5	0.057
5	R2	0.040
6	Hotel ID	0.031
7	SL4	0.029
8	ST6	0.028
9	C1	0.023

Only one social feature, the number of reviews submitted for the hotel (**SL4**), was ranked in the top 9 features. Similarly, only a single content feature, the number of terms in the review text (**C1**), was located in the top features. None of the features relating to well-formed review text (**C2** and **C3**) was ranked highly. The small number of highly-ranked social and content features was also reflected in the relatively low AUC scores achieved by these features (Figure 1(a)).

Further, none of the features relating to review completeness (**C4**, **C5** and **C6**) was a strong predictor of helpful reviews. This finding is surprising, given that we expected more complete reviews to be more informative and thus more helpful to users. It may be that users do not value the particular form of optional content that can be included in reviews, and focus instead on comments expressed in the main review text. We will consider content features afresh in future work by, for example, incorporating some of the lexical and other content-based features as used in related work (Section 2).

Finally, we examine classification performance when review instances were constructed using only the top 9 features as ranked by information gain, condition 'IG' in Figure 1(a). The results show that comparable performance was seen using this approach as to when all features were used. This finding suggests that JRip proved to be robust to the noise introduced by lower-ranked features.

4.2 Recommendation Results

Ultimately, the use of classification techniques are a means to enable the recommendation of reviews to a user. To the extent that reasonable classification performance has been obtained, we can be optimistic that this approach can provide a basis for high quality recommendations. In this section, we evaluate the quality of these recommendations.

We adopt the following form of recommendation. Taking our lead from Amazon as discussed above, our recommender selects two reviews per hotel: (1) the most helpful highly-scored ($\geq$ 4-stars) review and (2) the most helpful poorly-scored ($<$ 4-stars) review. Further, we consider two alternatives to our classification-based recommendation technique by ranking reviews by *date* (recommending the most recent highly-scored and poorly-scored reviews) and ranking reviews at *random* (recommending a randomly selected highly-scored and poorly-scored review).

Test sets are constructed from the balanced dataset using only those hotels which had received 5 or more highly-scored or poorly-scored reviews. There were 528 and 224 such hotels in the dataset, respectively. For each test set hotel, we

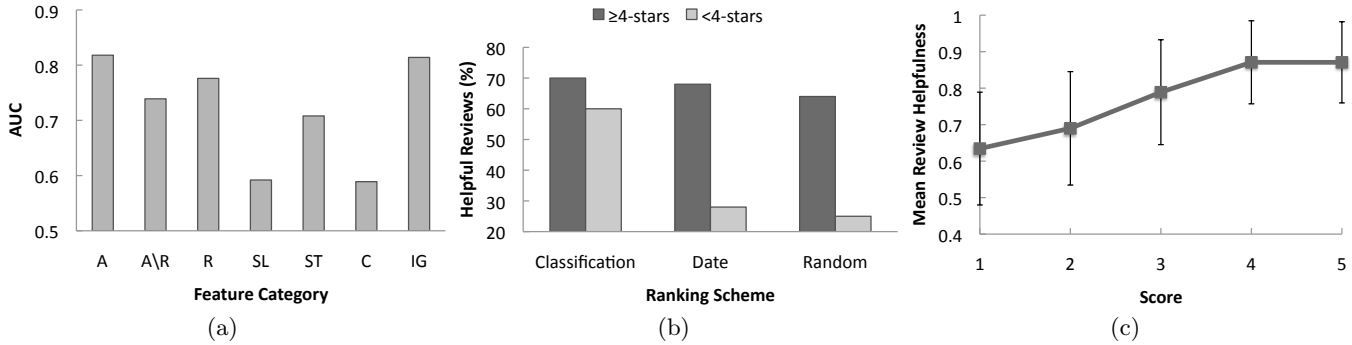

Figure 1: AUC for JRip versus feature category (a), percentage of helpful reviews in recommendations made over all test set hotels versus ranking scheme (b) and mean review helpfulness versus score (c)

recommend its most helpful highly-scored or poorly-scored review using JRip which is trained on the reviews of all other hotels in the dataset. In these experiments, training instances contain all features.

To evaluate recommendation performance, we consider how frequently the various recommenders manage to select a review that is unambiguously helpful according to our definition given in Section 3; that is, a review that has received at least 75% positive opinions. Ranking by *classification* provided very substantial benefits in relation to the recommendation of poorly-scored reviews. Using this approach, 60% of recommended reviews across test set hotels were helpful, compared to only 28% and 25% for *date* and *random*, respectively (Figure 1(b)). Our approach achieved much more modest improvements relative to *date* and *random* in the case of highly-scored reviews. This result can be attributed to the high average review helpfulness that was observed for such reviews (Figure 1(c)), and thus all three ranking schemes were able to achieve good performance.

5. CONCLUSIONS

The above findings demonstrate that our classification-based recommender achieved a high level of performance in terms of identifying and recommending the most helpful reviews. Interestingly, significantly better performance was seen for poorly-scored reviews. We believe that this result has importance given that the average helpfulness of poorly-scored reviews was relatively low (Figure 1(c)), and hence the need for a scheme that can accurately rank such reviews.

Reputation and sentiment features proved to be most useful in terms of classification performance. Social and content features were less successful. Classification performance remained high, however, in the absence of reputation features, which is important given that such features are not always available. In future work, we plan on incorporating additional review features in our analysis, such as those outlined in Section 2. Further, the classification-based recommender introduced in this paper is generalisable to other domains, an analysis of which we will also consider in future work.

6. REFERENCES

[1] S. Baccianella, A. Esuli, and F. Sebastiani. Multi-facet rating of product reviews. In *Advances in Information Retrieval, 31th European Conference on Information Retrieval Research (ECIR 2009)*, April 6–9 2009.

[2] M. Bilgic and R. J. Mooney. Explaining recommendations: Satisfaction vs. promotion. In *Beyond Personalization Workshop, held in conjunction with the 2005 International Conference on Intelligent User Interfaces*, San Diego, CA, USA, 2005.

[3] T. Fawcett. Roc graphs: Notes and practical considerations for researchers. In *Technical Report HPL-2003-4, HP Laboratories, CA, USA*, 2004.

[4] F. M. Harper, D. Moy, and J. A. Konstan. Facts or friends? distinguishing informational and conversational questions in social Q&A sites. In *Proceedings of the 27th International Conference on Human Factors in Computing Systems (CHI'09)*, pages 759–768, Boston, MA, USA, 2009.

[5] J. L. Herlocker, J. A. Konstan, and J. Riedl. Explaining collaborative filtering recommendations. In *Proceeding on the ACM 2000 Conference on Computer Supported Cooperative Work*, pages 241–250, Philadelphia, PA, USA, 2000.

[6] 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*, pages 423–430, Sydney, Australia, July 22–23 2006.

[7] 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.

[8] H. Tang, S. Tan, and X. Cheng. A survey on sentiment detection of reviews. *Expert Systems With Applications*, 36(7):10760–10773, 2009.

[9] N. Tintarev and J. Masthoff. The effectiveness of personalized movie explanations: An experiment using commercial meta-data. In *Proceedings of the 5th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH 2008)*, 2008.

[10] W. Weerkamp and M. de Rijke. Credibility improves topical blog post retrieval. In *Proceedings of the Association for Computational Linguistics with the Human Language Technology Conference (ACL-08:HLT)*, pages 923–931, June 16–18 2008.

[11] I. H. Witten and E. Frank. *Data Mining – Practical Machine Learning Tools and Techniques, 2nd Edition*. Elsevier, 2005.