Title	A classification-based review recommender
Authors(s)	O'Mahony, Michael P.; Smyth, Barry
Publication date	2010-05
Publication information	Knowledge-Based Systems, 23 (4): 323-329
Publisher	Elsevier
Link to online version	http://dx.doi.org/10.1016/j.knosys.2009.11.004
Item record/more information	http://hdl.handle.net/10197/2000
Publisher's version (DOI)	10.1016/j.knosys.2009.11.004

A Classification-based Review Recommender

M.P.O'Mahony*, B.Smyth

CLARITY: Centre for Sensor Web Technologies, School of Computer Science and Informatics, University College Dublin, Belfield, Dublin 4, Ireland

Abstract

Many online stores encourage their users to submit product or service reviews in order to guide future purchasing decisions. These reviews are often listed alongside product recommendations but, to date, limited attention has been paid as to how best to present these reviews to the end-user. In this paper, we describe a supervised classification approach that is designed to identify and recommend the most *helpful* product reviews. Using the TripAdvisor service as a case study, we compare the performance of several classification techniques using a range of features derived from hotel reviews. We then describe how these classifiers can be used as the basis for a practical recommender that automatically suggests the most-helpful contrasting reviews to end-users. We present an empirical evaluation which shows that our approach achieves a statistically significant improvement over alternative review ranking schemes.

Key words: user-generated reviews, classification, helpful, TripAdvisor

1. Introduction

Recommendations are now an established part of online life. In the so-called *Social Web*, we receive recommendations everyday from friends and colleagues, as well as from more distant connections in our growing social graphs. Recommender systems have played a key role in automating the generation of high-quality recommendations based on our online histories

*Principal corresponding author

Email addresses: michael.p.omahony@ucd.ie (M.P.O'Mahony), barry.smyth@ucd.ie (B.Smyth)

and/or purchasing preferences. For example, music services such as Pandora and Last.fm are distinguished by their ability to suggest interesting music based on our short-term and long-term listening habits. Indeed, online stores such as Amazon, iTunes, and BestBuy have long established the critical role of recommender systems when it comes to turning browsers into buyers.

Recently, information in the form of *user-generated reviews* has become increasingly important when it comes to helping users make the sort of buying decisions that recommender systems hope to influence. Many sites, such as Amazon, TripAdvisor and Yelp, complement their product descriptions with a rich collection of user reviews. Indeed, many of us use sites like Amazon and TripAdvisor primarily for their review information, even when we make our purchases elsewhere. In the world of recommender systems, reviews serve as a form of *recommendation explanation* (Bilgic and Mooney, 2005; Herlocker et al., 2000; Tintarev and Masthoff, 2008), helping users to evaluate the quality of suggestions.

The availability of user-generated reviews introduces a new type of recommendation problem. While reviews are becoming increasingly more common, they can vary greatly in their quality and helpfulness. For example, reviews can be biased or poorly authored, while others can be very balanced and insightful. For this reason, the ability to accurately identify helpful reviews would be a useful, albeit challenging, feature to automate. While some services are addressing this by allowing users to rate the helpfulness of each review, this type of feedback can be sparse and varied, with many reviews, particularly the more recent ones, failing to attract any feedback.

In this paper, we describe a system that is designed to recommend the most helpful product reviews to users. In the next section, we motivate the task in the context of the TripAdvisor service, which we use as a test domain. In Section 3, we adopt a classification approach to harness available review feedback to learn a classifier that identifies helpful and non-helpful reviews. We then describe how this classifier can be used as the basis for a practical recommender that automatically suggests the most-helpful contrasting reviews to end-users. In Section 4, we describe a comprehensive evaluation that is based on a large set of TripAdvisor hotel reviews. We show that our recommender system is capable of suggesting superior reviews compared to benchmark approaches, and highlight an interesting performance asymmetry that is biased in favour of reviews expressing negative sentiment.

2. Towards Recommending Helpful Reviews

Insightful product reviews can be extremely helpful in guiding purchasing decisions. As reviews accumulate, however, it can become difficult for users to identify those that are helpful, thereby introducing yet another information overload problem. This signals a new and challenging recommendation task—to recommend reviews based on *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 users, and the role of the *review recommender* is to suggest a small number of helpful reviews for each of these products. We address review recommendation in Section 2.3, but first we consider user-generated reviews and review helpfulness in respect of TripAdvisor reviews, which form the basis of our study.

2.1. TripAdvisor Reviews

Figure 1 shows a typical TripAdvisor review. In addition to the *hotel ID* and the *user ID*, each review includes an overall *score* (in this example, 5 out of 5 stars), a *title* ("The Best Place") and the *review-text* (in this case, just three lines of text).

Optionally, users can specify what they *liked* and *disliked* 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 (*Your age range* and *Member since*) and details relating to the date and purpose of visit (*Date of Stay*, *Visit was for* and *Traveling group*). Finally, users can respond to set review-template questions such as *Would I recommend this hotel to my best friend?* and *I recommend this hotel for*.

For the study described in this paper, we created two large datasets by extracting all TripAdvisor reviews prior to April 2009 for users who had reviewed any hotel in either of two US cities, Chicago or Las Vegas. In total, there are approximately 225,000 reviews by 45,000 users on 70,000 hotels (Table 1). For both datasets, the median number of reviews per user and per hotel is 7 and 1, respectively. These distributions are, however, significantly skewed; for example, the most reviewed hotel in the Chicago and Las Vegas datasets has 575 and 2205 reviews, respectively, while the greatest number of reviews written by any user is 165 and 134, respectively.

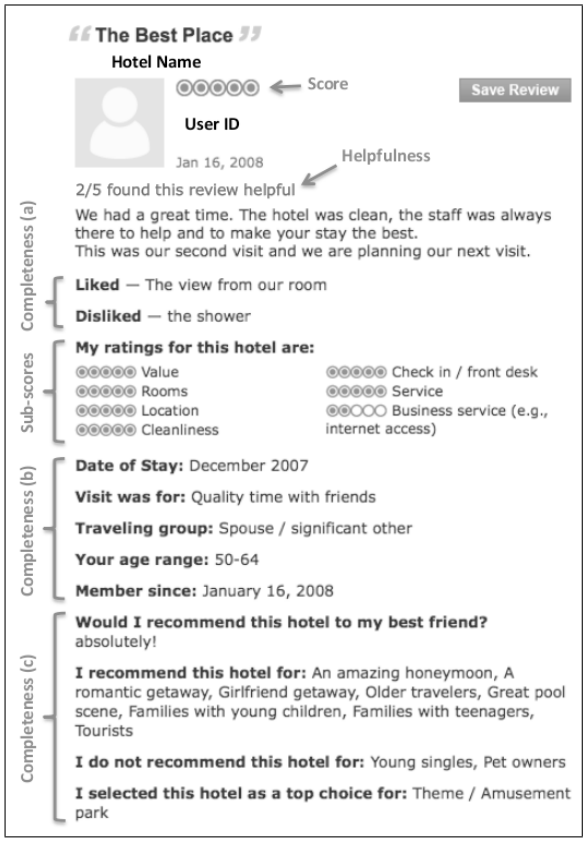


Figure 1: A TripAdvisor review

2.2. Review Helpfulness

Importantly for our case study, TripAdvisor allows users to provide feedback on review helpfulness. We define *helpfulness* as the percentage of positive *opinions* that a review has received. For example, the review shown in Figure 1 has received 2 positive and 3 negative opinions and thus it has a helpfulness of 0.4.

Figure 2(a) shows the number of reviews in the Las Vegas dataset versus user score. It is clear that the majority of reviews attracted high scores, with more than 95,000 4-star and 5-star reviews submitted, compared to less than 10,000 1-star reviews. This suggests that users are far more likely to review hotels that they have liked.

In addition, Figure 2(a) indicates that many reviews attracted very few opinions; for example, approximately 20% of reviews received no feedback

Table 1: TripAdvisor dataset statistics

Dataset	# Users	# Hotels	# Reviews
Chicago	13,473	28,840	77,863
Las Vegas	32,002	41,154	146,409

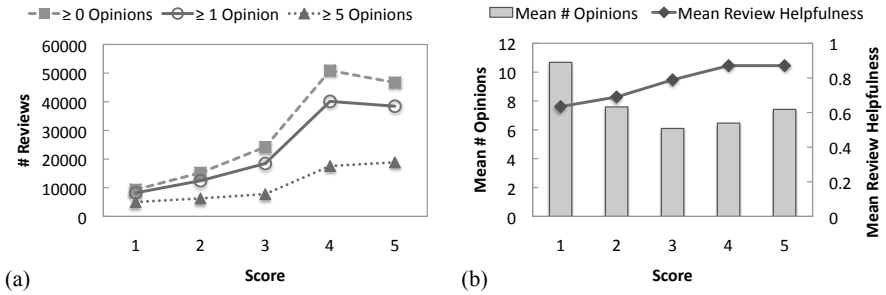


Figure 2: Las Vegas dataset trends. **a** number of reviews versus score. **b** mean number of opinions per review and mean review helpfulness versus score. Similar trends applied for the Chicago dataset

and, while some 80% of reviews received ≥ 1 opinion, only 38% of reviews received ≥ 5 opinions. Excluding reviews with no feedback, Figure 2(b) shows that the most poorly-scored reviews attracted on average the highest number of opinions (almost 11), while reviews with scores of $\geq$ 2-stars received on average between 6 and 8 opinions.

Interestingly, reviews with lower scores were perceived as being less helpful by users (Figure 2(b)). For example, on average 63% of opinions for 1-star reviews (approximately 7 out of 11 opinions) were positive, with 4 out of 11 opinions being negative. In contrast, of the 7 opinions attracted by 5-star reviews, 87% were positive; thus, only about 1 of 7 opinions attracted by such reviews were negative. In other words, 1-star reviews attracted on average almost 3 times as many negative opinions as 5-star reviews, indicating that users were far more divided in their judgements about the helpfulness of reviews with low scores compared to those with high scores.

2.3. Review Ranking Schemes

The above findings indicate that relying on review feedback alone to recommend and rank reviews is insufficient, given that many reviews fail to attract the critical mass of opinions that would permit reliable helpfulness assessments to be made. Currently, TripAdvisor ranks reviews either by date

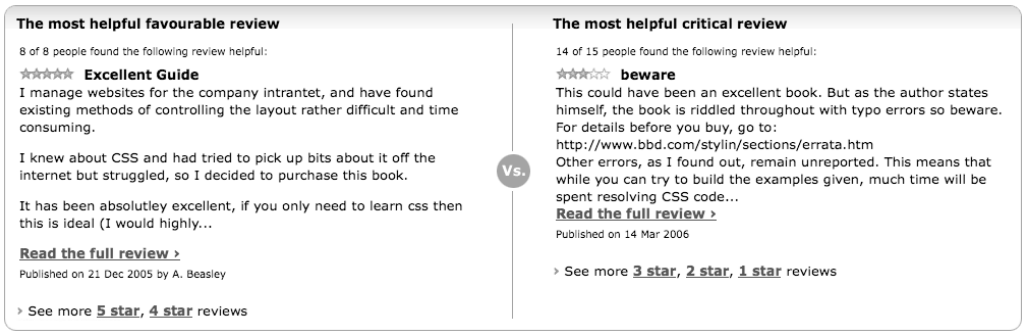


Figure 3: Review ranking: an example of the Amazon.com approach of listing the most helpful poorly-scored and highly-scored product reviews side by side

or by user score, but there is no guarantee that the most recent or highly-scored reviews are the most helpful.

In other domains, more sophisticated approaches to review ranking have been explored. For example, Amazon now suggest the most helpful poorly-scored and highly-scored reviews alongside summary product descriptions (Figure 3). From a review recommendation standpoint, we believe this is a step in the right direction, as it enables users to rapidly assess product quality. Again, however, this approach is limited to cases where sufficient feedback on review helpfulness had been amassed.

The main objective of this paper is to develop a classification approach to identify the most helpful product reviews. Our approach, which is detailed in the next section, 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 received no feedback on review helpfulness. Indeed, such a classifier may be generalisable to domains where review helpfulness data is not collected; although this question we leave for future work.

3. Classifying and Recommending Reviews

We adopt a supervised approach to classifying the most helpful reviews. Review instances are labeled as *helpful* or *non-helpful*, and a review is considered *helpful* if and only if at least 75% of opinions for the review are positive. In this way we focus the classification task on the prediction of the most unambiguously helpful reviews.

3.1. Classification Features

Prior to classification, each review is translated into a feature-based instance representation. Review instances consist of features derived from four distinct categories which are mined from individual reviews and from the wider community reviewing activity. We refer to these categories as *user reputation* (R), *social* (SL), *sentiment* (ST) and *content* (C). Thus each review instance, I_j, can be expressed as follows: $I_j = \{R_j, SL_j, ST_j, C_j, class_j\}$, where $class_j = \{\text{helpful, non-helpful}\}$ as described above. In the following sections, the feature categories are described in turn.

3.1.1. User Reputation Features

These 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:

R1: The mean review helpfulness over all reviews authored by the user.

R2: The standard deviation of review helpfulness over all reviews authored by the user.

R3: The percentage of reviews authored by the user which have received a minimum of T opinions; in this work, $T = 5$ (see Section 4.1).

These features capture the intuition that users who authored helpful reviews in the past are likely to do so in future. As such, we expect reputation features to be strong predictors of review helpfulness. Given that many reviews, however, receive only limited feedback on review helpfulness, we explicitly evaluate classifier performance when reputation features are excluded from review instances in Section 4.

3.1.2. Social Features

These features are concerned with the degree distribution in the bipartite user–hotel review graph. We mine six such features in total from our datasets, which are:

SL1: The number of reviews authored by the user.

SL2: The mean number of reviews authored by all users.

SL3: The standard deviation of the number of reviews authored by all users.

SL4: The number of reviews submitted for the hotel.

SL5: The mean number of reviews submitted for all hotels.

SL6: The standard deviation of the number of reviews submitted for all hotels.

The above features can be considered as a kind of "rich get richer" phenomenon where, for example, reviews authored by more experienced reviewers may have improved quality. It is uncertain if this concept of experience applies to hotels; however, our rationale for the latter three features is that when users write their own reviews they may, for example, respond to comments made in existing reviews submitted for a particular hotel, and thereby improve the quality of their own reviews.

3.1.3. Sentiment Features

Sentiment relates to how well users enjoyed their experience with a hotel. In this paper, we consider sentiment in terms of the score (expressed on a 5-star scale) that the user has assigned to a hotel. In addition, we consider the optional sub-scores that may be assigned by users (see Section 2.1). We extract the following features:

ST1: The score assigned by the user to the hotel.

ST2: The number of (optional) sub-scores assigned by the user.

ST3: The mean sub-score assigned by the user.

ST4: The standard deviation of the sub-scores assigned by the user.

ST5: The mean score over all reviews authored by the user.

ST6: The standard deviation of the scores over all reviews authored by the user.

ST7: The mean score assigned by the all users to the hotel.

ST8: The standard deviation of scores assigned by the all users to the hotel.

The analysis presented in Section 2.2 indicated that score was indeed an indicator of review helpfulness, where highly-scored reviews attracted the greatest number of positive helpfulness opinions. The importance of sentiment features from a classification perspective is examined in further detail in the evaluation section (Section 4).

3.1.4. Content Features

We consider several features in respect of review content:

C1: The number of terms in the review text.

C2: The ratio of uppercase and lowercase characters to other characters in the review text.

C3: The ratio of uppercase to lowercase characters in the review text.

C4: Review completeness (a) – an integer in the range [0,2] which captures whether the user has completed one, both or none of the optional *liked* and *disliked* parts of the review (see Section 2.1).

C5: Review completeness (b) – the number of optional personal and purpose of visit details that are provided by the user in the review (see Section 2.1).

C6: Review completeness (c) – the number of optional review-template questions that are answered in the review (see Section 2.1).

The first feature is designed to distinguish between reviews based on the length of the review-text. The second and third features are intended to capture whether or not the review text is well formed; for example, the absence of uppercase or punctuation characters is indicative of a poorly authored review, and such reviews are unlikely to be perceived favourably by users. The final three features provide a measure of review completeness, i.e. how much optional content has been included in reviews. We expect that more complete reviews are likely to be more helpful to users.

3.2. Recommendation via Classification

We can use our collection of review instances as supervised training data for a variety of standard classification algorithms. In this paper we consider the JRip, J48, and naïve Bayes classifiers (Witten and Frank, 2005). Each

technique produces a *classifier* from the training data which can be used to classify unseen instances (reviews) in the absence of helpfulness data. In addition, each classifier can return not just the predicted class (helpful vs. non-helpful), but also a *confidence* score for the associated prediction.

Prediction confidence can then be used to effectively translate review classification into review recommendation, by rank-ordering reviews classified as helpful according to their prediction confidence. Thus, given a set of reviews for a hotel, we can use a classifier to produce a ranked list of those reviews predicted to be helpful.

Other recommendation styles are also possible; for example, the Amazon approach of recommending the most-helpful highly-scored and poorly-scored product reviews to provide the user with contrasting reviews (Figure 3). All such approaches are, however, 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 recommend reviews that have not attracted any (or a critical mass of) feedback.

4. Evaluation

So far we have motivated the need for *review recommendation* as a complement to *product recommendation*. We have described how a classification approach can be adopted as a basis for recommendation. Ultimately, success will depend on classification accuracy and how this translates into useful recommendations. We now examine these issues in the context of a large-scale study using TripAdvisor data.

4.1. Datasets and Methodology

To provide training data for the classifiers, features were first computed over all review instances in the Chicago and Las Vegas datasets. To provide support when labeling reviews, we selected only those reviews with a minimum of $T = 5$ 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. Table 2 shows the statistics for these balanced datasets.

Following Harper et al. (2009), we report *sensitivity* and *specificity*, which measure the proportion of helpful and non-helpful reviews that are correctly classified, respectively. In addition, we report AUC (area under ROC curve)

Table 2: Balanced dataset statistics

Dataset	# Users	# Hotels	# Reviews
Chicago	7,399	7,646	17,038
Las Vegas	18,849	10,782	35,802

which produces a value between 0 and 1; higher values indicate better classification performance. Further details on these metrics can be found in Fawcett (2004). The relative performance of the JRip, J48, and naïve Bayes (NB) classification techniques are compared using Weka (Witten and Frank, 2005).

4.2. Classification Results

Classification performance is measured using a standard 10-fold cross-validation technique. In the following sections we describe the classification results obtained across different groupings of features and feature types.

4.2.1. Classification using All Features

We begin by examining classification performance when all available features (that is, reputation, social, sentiment, content plus three generic features: *user-id*, *hotel-id* and *review date*) are considered. The sensitivity, specificity, and AUC results are presented in Figure 4, as the bars labeled 'A' for the Chicago and Las Vegas datasets. Overall, JRip was seen to outperform J48 and NB for both datasets and across all evaluation metrics. In addition, J48 usually performed better than NB.

Reputation features include information about the helpfulness of other reviews authored by the review author, and for this reason they are likely to be influential. Thus we have also included results for training instances that include all features except reputation features, condition 'A\R' in Figure 4. As expected, we see a drop in classification performance across the datasets and algorithms suggesting that reputation features do in fact play an important role. We will return to this point in the next section, but for now we highlight that even in the absence of reputation features — and remember that these features are not available in all domains — classification performance remains high for both datasets with AUC scores > 0.72 for JRip.

4.2.2. Classification by Feature Category

The performance of classifiers trained using the reputation, social, sentiment, and content feature categories are also presented in Figure 4, as bars

Table 3: Features ranked by information gain (IG)

	Chicago		Las Vegas	
Rank	Feature ID	IG	Feature ID	IG
1	R1	0.085	R1	0.172
2	Hotel ID	0.077	ST1	0.095
3	SL4	0.052	ST3	0.079
4	SL1	0.051	ST5	0.057
5	ST1	0.047	R2	0.040
6	R2	0.045	Hotel ID	0.031
7	ST5	0.045	SL4	0.029
8	ST6	0.044	ST6	0.028
9	ST3	0.043	C1	0.023

labeled 'R', 'SL', 'ST' and 'C', respectively. The results highlight the strong performance of the reputation features in particular. For example, the AUC metric clearly shows that reputation features provided the best performance, followed by sentiment features. In the case of the Las Vegas dataset, for example, the best performing classifier (J48) achieved AUC scores of 0.82 and 0.73 using reputation and sentiment features, respectively. Both social and content feature sets were less successful, with J48 achieving AUC scores of 0.60 and 0.61, respectively. Broadly similar trends were observed for the sensitivity and specificity metrics. In most cases, higher sensitivity rates were achieved, which indicates that more false positives were seen than false negatives.

4.2.3. 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 3 the top 9 features for both datasets, 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 for both datasets. Overall, we find that 8 out of 9 features were common across both datasets, albeit with different rank orderings. More or less the same groupings of reputation, social and sentiment features were found, with social features proving to be more important (in terms of rank) than sentiment features in the Chicago

dataset, and vice versa for Las Vegas.

In relation to social features, both the number of reviews submitted for the hotel (**SL4**) and the number of reviews written by the user (**SL1**) were among the top ranked features for the Chicago dataset, although only one (**SL4**) was ranked in the top 9 in the Las Vegas dataset. A total of 4 sentiment features (**ST1**, **ST3**, **ST5** and **ST6**) are ranked among the top features for both datasets (although in different order), reflecting the relatively good classification performance achieved by such features as shown in Figure 4. In particular, the importance of **ST1** (the score assigned by the user to the hotel) was previously discussed in Section 2.2. The power of this feature is further indicated in terms of information gain: for both datasets, **ST1** was the highest ranked sentiment feature, and was ranked 2nd and 5th for the Las Vegas and Chicago datasets, respectively.

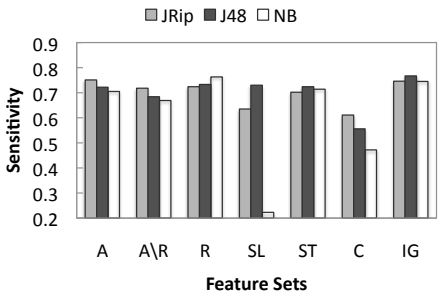
Only a single content feature, the number of terms in the review text (**C1**), was located in the top features for one of the datasets (Las Vegas). None of the features relating to well-formed review text (**C2** and **C3**) were ranked highly. Further, none of the features that indicate review completeness (**C4**, **C5** and **C6**) were strong predictors of helpful reviews. It remains an open question as to why content features were not particularly useful predictors of review helpfulness. A more comprehensive analysis in respect of review content is certainly possible (see Section 5); we will consider content features afresh in future analysis.

Finally, we examine classification performance when review instances were constructed using only the top 9 features as ranked by information gain. The results in Figure 4 (condition 'IG') show that best AUC performance was seen for both datasets using J48 with this approach. This finding suggests that the low information gain associated with the remaining features essentially introduced noise into the classification process and that their removal lead to an improvement in overall performance.

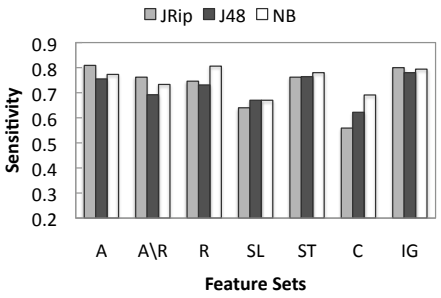
4.3. Recommendation Results

Ultimately, classification techniques are a means to enable the recommendation of reviews to users. 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. We now evaluate the quality of these recommendations.

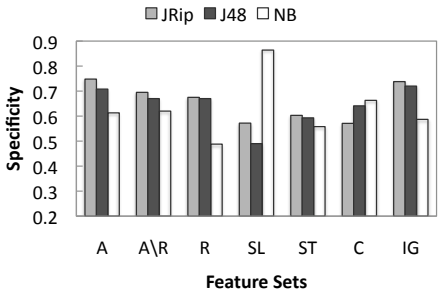
We adopt the following form of recommendation. Taking our lead from Amazon as discussed above, our recommender selects two reviews per hotel:



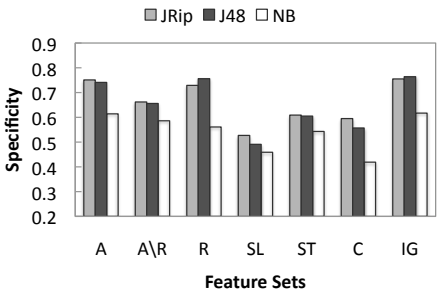
(a) Chicago – Sensitivity



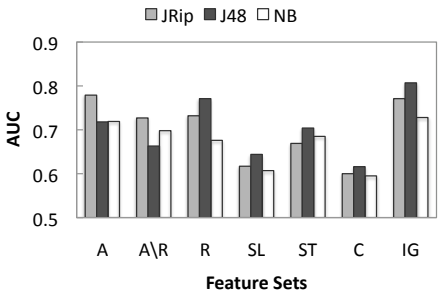
(b) Las Vegas – Sensitivity



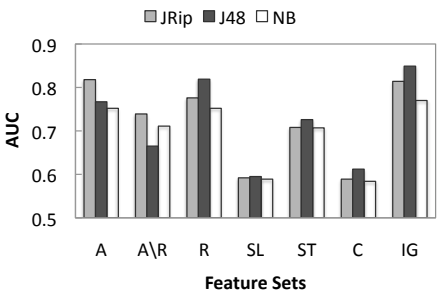
(c) Chicago – Specificity



(d) Las Vegas – Specificity



(e) Chicago – AUC



(f) Las Vegas – AUC

Figure 4: Classification performance for the Chicago and Las Vegas datasets. Notation: A – all features, A\R – all excluding reputation features, R – reputation features, SL – social features, ST – sentiment features, C – content features and IG – top-9 features ranked by information gain

(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).

Recommendation test sets are constructed from the balanced datasets using only those hotels which have a minimum of 5 highly-scored or poorly-scored reviews. In the Chicago dataset, there are 239 and 124 hotels with 5 or more highly-scored and poorly-scored reviews, respectively. In the Las Vegas dataset, there are 528 and 224 such hotels, respectively. We adopt a leave-one-out recommendation approach such that, for each test set hotel, we recommend its most helpful highly-scored or poorly-scored review using a JRip classifier 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 two related metrics. First we look at the average helpfulness of recommended reviews produced by the different recommendation techniques. Results are shown in Table 4(a). Interestingly, the classification-based approach provided greatest benefit in relation to the recommendation of poorly-scored reviews. For example, for the Chicago dataset we see that the classification-based technique recommended such reviews with an average helpfulness of 0.71, compared to only 0.58 for *date* and *random*; an even greater benefit was observed for the Las Vegas results. ANOVA and Tukey HSD tests indicated that these differences were statistically significant at the $p < .01$ level. Thus we can conclude that the classification approach significantly outperformed the other two ranking schemes in terms of recommending the most helpful poorly-scored reviews.

The classification approach achieved more modest improvements for highly-scored reviews. The pairwise differences in means between the classification approach and the two other strategies were statistically significant at the $p < .05$ and $p < .01$ levels, respectively, for the Chicago dataset. No significant differences between ranking schemes were found, however, for the Las Vegas dataset. This finding can be attributed to the high average review helpfulness that highly-scored reviews generally attracted (see Figure 2(b)), and thus the *date* and *random* ranking schemes were able to achieve comparable performance to the classification-based approach.

As a second evaluation metric, we consider how frequently our recom-

Table 4: Average recommendation performance over test set hotels. **a** average review helpfulness of recommended reviews. **b** percentage of helpful reviews in recommended reviews

(a)

	Chicago			Las Vegas		
Score	Class.	Date	Rnd.	Class.	Date	Rnd.
≥4-stars	0.83	0.79	0.75	0.82	0.81	0.81
<4-stars	0.71	0.58	0.58	0.76	0.56	0.60

(b)

	Chicago			Las Vegas		
Score	Class.	Date	Rnd.	Class.	Date	Rnd.
≥4-stars	79	62	54	70	68	64
<4-stars	50	27	17	60	28	25

menders 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. The results are presented in Table 4(b). Overall, the trends are similar to those reported above, with the classification approach achieving the greatest improvements in the case of poorly-scored reviews. The results are also of interest, however, in relation to highly-scored reviews, since they indicate how even small changes in average review helpfulness translate into more significant recommendation improvements: many more unambiguously helpful reviews are recommended by the classification approach when compared to *date* and *random*. For example, in the case of the Chicago dataset, a percentage improvement of 5% in average review helpfulness from 0.79 (ranking by *date*) to 0.83 (ranking by classification) results in a relative improvement of 27% (from 62% to 79%) in the actual number of helpful reviews that are recommended. As expected, much smaller improvements were seen for the Las Vegas dataset, given that no statistically significant differences in average review helpfulness between ranking schemes were indicated for this dataset.

5. Discussion and Conclusions

The findings of Section 4 demonstrate that our approach achieved a high level of performance in terms of classifying and recommending the most help-

ful reviews. Greater performance was observed for poorly-scored reviews, which is significant given that these reviews were perceived on average as being less helpful by users, and hence the need for a scheme which can effectively rank such reviews. Overall, our findings are encouraging, taking into consideration that review helpfulness is a subjective notion and that many factors can influence user opinion in this regard.

There is rich scope for future work in this area and the following related work is of interest. A similar approach to review classification has been proposed by Kim et al. (2006), which considered feature sets relating to the structural, lexical, syntactic, semantic and some meta-data properties of reviews. Of these features, score, review length and unigram (term distribution) were among the most discriminating. (This work did not consider social or reputation features.) Reviewer expertise was found to be a useful predictor of review helpfulness by Liu et al. (2008), capturing the intuition that people interested in a particular genre of movies are likely to author high quality reviews for movies within the same or related genres. Timeliness of reviews was also important, and it was shown that (movie) review helpfulness declined as time went by.

A classification approach was applied by Harper et al. (2009) to distinguish between conversational and informational questions in social Q&A sites. In this work, features such as question category, text categorization and social network metrics were selected as the basis for classification and good performance was achieved. An analysis of credibility indicators in relation to topical blog post retrieval was presented by Weerkamp and de Rijke (2008). Some of the indicators considered were text length, the appropriate use of capitalisation and emoticons in the text, spelling errors, timeliness of posts and the regularity at which bloggers post. The use of such indicators was found to significantly improve retrieval performance by Weerkamp and de Rijke (2008). Work in relation to sentiment and opinion analysis (Tang et al., 2009) is also of interest. For example, the classification of reviews for sentiment using content-based feature sets was considered by Baccianella et al. (2009), where a study based on TripAvisor reviews demonstrated the effectiveness of this approach. Additional related work can be found in (O'Mahony and Smyth, 2009; O'Mahony et al., 2009; Hsu et al., 2009).

The framework introduced in this paper for the classification and recommendation of reviews is generalisable to other domains. In future work, we will apply our approach to review sites such as Amazon and Blippr; the

classification of reviews from the latter site in particular pose new challenges, given that reviews in this domain are constrained to 160 characters in length. In addition, motivated by the above related work, we will explore the use of richer sets of review features in our analysis.

6. Acknowledgements

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

References

Baccianella, S., Esuli, A., Sebastiani, F., 2009. Multi-facet rating of product reviews. In: Advances in Information Retrieval, 31th European Conference on Information Retrieval Research (ECIR 2009). Springer, Toulouse, France, pp. 461–472.

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

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

Harper, F. M., Moy, D., Konstan, J. A., 2009. 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 2009). Boston, MA, USA, pp. 759–768.

Herlocker, J. L., Konstan, J. A., Riedl, J., 2000. Explaining collaborative filtering recommendations. In: Proceeding on the ACM 2000 Conference on Computer Supported Cooperative Work (CSCW 2000). Philadelphia, PA, USA, pp. 241–250.

Hsu, C.-F., Khabiri, E., Caverlee, J., 2009. Ranking comments on the social web. In: Proceedings of the 2009 IEEE International Conference on Social Computing (SocialCom-09). Vancouver, Canada, pp. 90–97.

Kim, S.-M., Pantel, P., Chklovski, T., Pennacchiotti, M., 2006. Automatically assessing review helpfulness. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2006). Sydney, Australia, pp. 423–430.

Liu, Y., Huang, X., An, A., Yu, X., 2008. Modeling and predicting the helpfulness of online reviews. In: Proceedings of the 2008 Eighth IEEE International Conference on Data Mining (ICDM 2008). IEEE Computer Society, Pisa, Italy, pp. 443–452.

O'Mahony, M. P., Cunningham, P., Smyth, B., 2009. An assessment of machine learning techniques for review recommendation. In: Proceedings of the 20th Irish Conference on Artificial Intelligence and Cognitive Science (AICS 2009). Dublin, Ireland, pp. 244–253.

O'Mahony, M. P., Smyth, B., 2009. Learning to recommend helpful hotel reviews. In: Proceedings of the 3rd ACM Conference on Recommender Systems (RecSys 2009). New York City, NY, USA.

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

Tintarev, N., Masthoff, J., 2008. 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). Hannover, Germany, pp. 204–213.

Weerkamp, W., de Rijke, M., 2008. Credibility improves topical blog post retrieval. In: Proceedings of the Association for Computational Linguistics with the Human Language Technology Conference (ACL-08:HLT). Columbus, Ohio, USA, pp. 923–931.

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