Great Explanations: Opinionated Explanations for Recommendations

Khalil Muhammad(✉), Aonghus Lawlor, Rachael Rafter, and Barry Smyth

Insight Centre for Data Analytics, School of Computer Science and Informatics,
University College Dublin, Dublin, Ireland
khalil.muhammad@insight-centre.org

Abstract. Explaining recommendations helps users to make better decisions. We describe a novel approach to explanation for recommender systems, one that drives the recommendation ranking process, while at the same time providing the user with useful insights into the reason why items have been recommended and the trade-offs they may need to consider when making their choice. We describe this approach in the context of a case-based recommender system that harnesses opinions mined from user-generated reviews, and evaluate it on TripAdvisor hotel data.

Keywords: Recommender systems · Case-based reasoning · Explanations · Opinion mining · Sentiment analysis

1 Introduction

Recommender systems are a familiar part of the digital landscape helping millions of users make better choices about what to watch, wear, read, and buy. But generating suggestions is just the start. Explaining recommendations can make it easier for users to make decisions, increasing conversion rates and leading to more satisfied users [1–5]. Usually explanations provide a post-hoc rationalisation for the suggested items. But our work is motivated by a more intimate connection between recommendations and explanations, which poses the question: can the recommendation process itself be guided by structures generated to explain the suggestions to users?

We describe a case-based hotel recommender based on cases that are mined from the opinions in user-generated reviews; see also [6–8]. The central contribution of this work is a technique for generating personalised, feature-based explanations that can be used as part of an explanation interface in a recommender system but also during recommendation ranking. We provide examples based on real-world TripAdvisor data and discuss the results of an initial evaluation to explore the structure and utility of the resulting explanations.

2 Related Work

There is a history of using explanations to support reasoning in intelligent systems with approaches based on heuristics [9], CBR [10–12], and model-based

E. Hüllermeier and M. Minor (Eds.): ICCBR 2015, LNAI 9343, pp. 244–258, 2015.
DOI: 10.1007/978-3-319-24586-7_17

techniques [13] for example. More recently explanations have been used to support the recommendation process ([1–5]) by justifying recommendations to users. Good explanations promote trust and loyalty, increase satisfaction, and make it easier for users to find what they want.

Early work explored the utility of explanations in collaborative filtering systems with [1] reviewing different models and techniques for explanation based on MovieLens data. They considered a variety of explanation interfaces leveraging different combinations of data (ratings, meta-data, neighbours, confidence scores etc.) and presentation styles (histograms, confidence intervals, text etc.) concluding that most users recognised the value of explanations.

Bilgic and Mooney [14] used keywords to justify items rather than disclosing the behaviour of similar users. They argued that the goal of an explanation should not be to "sell" the user on the item but rather to help the user to make an informed judgment. They found users tended to overestimate item quality when presented with similar-user style explanations. Elsewhere, keyword approaches were further developed by [2] in a content-based, collaborative hybrid recommender capable of providing explanations such as: *"Item A is suggested because it contains features X and Y that are also included in items B, C, and D, which you have also liked."*; see also the work of [15] for related ideas based on user-generated tags instead of keywords. Note that this style of explanation justifies the item with reference to other items, in this case items that the user had previously liked.

Explanations can also relate one item to others. For example, Pu and Chen [3] build explanations that emphasise the tradeoffs between items. For example, a recommended item can be augmented by an explanation that highlights alternatives with different tradeoffs such as *"Here are laptops that are cheaper and lighter but with a slower processor"* for instance; see also related work by [16].

Here we focus on generating explanations that are feature-based and personalized (see also [17]), highlighting features that are likely to matter most to the user. But, like the work of [3,16], our explanations also relate items to other recommendation alternatives to help the user to better understand the trade-offs and compromises that exist within a product-space; see also [18]. However, our work also leverages the opinions in user-generated reviews as its primary source of item and recommendation knowledge. A unique feature of our approach is that explanations are not generated purely to justify recommendations but also to influence their ranking in the recommendation set.

3 Mining Experiential Cases

Our approach is summarised in Fig. 1 which we will describe with reference to TripAdvisor hotels and reviews. The *opinion mining* component extracts features and sentiments from reviews to produce hotel cases. This also generates user profiles from the reviews a user has submitted (or, for example, from the reviews they have previously viewed or marked as useful). The recommendation engine takes a user query (and profile) and retrieves a set of matching hotels and

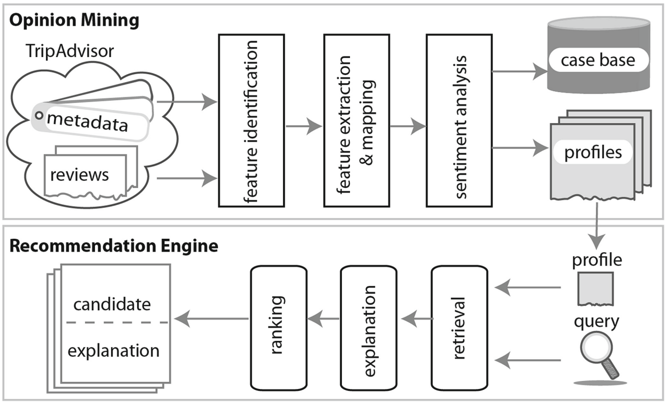


Fig. 1. An overview of the experiential product recommendation architecture.

then, generating explanations for each of these candidates, uses these explanations to rank the hotels for recommendation. It is this combination of opinion mining and explanation-based ranking that sets this work apart from others.

3.1 Opinion Mining

To identify and extract features from reviews we use the methods of [7,8]; we will refer to these (e.g. the *carpets* or the quality of *orange juice* at breakfast) as *review features*. While [7] use these as the basis for case descriptions, we find that they are less suitable for our needs, especially as the basis of explanations. For this reason we harness higher-level features available in the meta-data for hotels and map the review features back to these higher-level features. Since we will be focusing on TripAdvisor data, we map these review features back to a set of known *amenities* (e.g. *room quality*, *bar/restaurant* etc.); we refer to these features as *item features*. In this way we use this amenity meta-data as the primary features of our cases while still leveraging the opinions expressed in reviews to associate sentiment information with these amenities.

Mining Review Features. As with [8] we mine *bi-gram* features and *single-noun* features; see also [19,20]. For example, bi-grams which conform to one of two basic part-of-speech co-location patterns are considered — a noun followed by a noun, such as *shower screen* (NN), or an adjective followed by a noun, such as *twin room* (AN) — excluding bi-grams whose adjective is a sentiment word (e.g. *excellent, terrible* etc.) in the sentiment lexicon [19]. Separately, single-noun features are validated by eliminating nouns that are rarely associated with sentiment words in reviews as per [19], since such nouns are unlikely to refer to product features; these extracted features are the *review features*.

Mapping Review Features to Item Features. Taking all review texts, we apply k-means clustering, using sentence co-occurence, to associate review features with item features (amenities). While beyond the scope of this work suffice it to say that this provides a mapping between review features, such as *orange juice* and item features such as *breakfast*.

Evaluating Feature Sentiment. Again, as per [7], for a review feature f_i in a review sentence S_j, we determine whether there are any sentiment words in S_j. If not, f_i is marked *neutral*, otherwise we identify the sentiment word w_{min} with 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*. We compute the frequency of all opinion patterns recorded after a pass of all reviews; a pattern is *valid* if it occurs more than average. For valid patterns we assign sentiment to f_i based on the sentiment of w_{min} and subject to whether S_j contains any negation terms within 4 words of w_{min}. If there are no negation terms then the sentiment assigned to f_i in S_j is that of the sentiment word in the sentiment lexicon; otherwise this sentiment is reversed. If an opinion pattern is not valid then we assign a *neutral* sentiment to each of its occurrences within the review set; see [21] for a fuller description.

Generating Experiential Cases. For each item/hotel H_j we have review features $\{f_1, ..., f_m\}$ mined from $reviews(H_j)$. Each f_i is mapped to a item feature F_i and we aggregate the review feature's mentions and sentiment scores to associate them with the corresponding F_i. So $F(H_j)$ is the set of item features $\{F_1, ..., F_n\}$ of hotel H_j. We can compute various properties of F_i: the fraction of times it is mentioned in reviews (its *importance*, see Eq. 1) and the degree to which it is mentioned in a positive or negative light (its *sentiment*, see Eq. 2, where $pos(F_i, H_j)$ and $neg(F_i, H_j)$ denote the number of times that feature F_i has positive or negative sentiment in reviews for H_j, respectively). Thus, each hotel can be represented as a *case*, $case(H_j)$, which aggregates item features, importance and sentiment data as in Eq. 3.

$$imp(F_i, H) = \frac{count(F_i, H)}{\sum_{\forall F_k \in F(H_j)} count(F_k, H_j)} \tag{1}$$

$$sent(F_i, H_j) = \frac{pos(F_i, H_j)}{pos(F_i, H_j) + neg(F_i, H_j)} \tag{2}$$

$$case(H_j) = \{[F_i, sent(F_i, H_j), imp(F_i, H_j)] : F_i \in F(H_j)\} \tag{3}$$

3.2 The Recommendation Engine

The recommendation engine returns a set of items (hotels) based on some query and user profile. Previous work has described related approaches to recommendation using opinions and sentiment [6,7] but here we describe a very different approach, one that bases recommendation on the ability to generate compelling explanations. The core of this is a novel approach to generating opinionated explanations and a way to score these explanations for recommendation ranking. We will discuss this in detail in the next section of this paper.

4 Generating Opinionated Explanations

Before describing our explanation approach it is important to understand the setting: we assume the target user U_T is presented with a set of hotel recommendations $\{H_1...H_k\}$ based on some user query which might include features such as star rating, price and location, and our task is to generate an explanation for each H_i. To simplify the explanation process let us say for now that we will build an explanation that will highlight two types of features: (1) reasons why they might choose the hotel; and (2) reasons why they might avoid the hotel.

4.1 A Basic Explanation Structure

Our basic explanation comes in two parts. The *pro* part is a set of (positive) hotel features that are reasons to choose the hotel. The *con* part is a set of (negative) features that can be considered as reasons to avoid the hotel. More formally, a feature F_i of hotel H_T is a *pro* if and only if it has a majority of positive sentiments ($sent(F_i, H_T) > 0.7$ in the case of our TripAdvisor data) and if its sentiment is *better than* at least one of the alternative hotels, H' (that is, $betterThan(F_i, H_T, H') > 0$); see Eqs. 4 and 5. Obviously this does not guarantee a pro will be a strong reason to choose H_T — it might only be better than a small fraction of the alternatives — but it is a possible reason to choose the hotel. Likewise a feature is a *con* if it has a negative sentiment ($sent(F_i, H_T) < 0.7$) and if it is *worse than* at least one alternative case; see Eqs. 6 and 7.

$$pro(F_i, H_T, H') \leftrightarrow sent(F_i, H_T) > 0.7 \wedge betterThan(F_i, H_T, H') > 0 \tag{4}$$

$$betterThan(F_i, H_T, H') = \frac{\sum_{H_c \in H'} 1[sent(F_i, H_T) > sent(F_i, H_c)]}{|H'|} \tag{5}$$

$$con(F_i, H_T, H') \leftrightarrow sent(F_i, H_T) <= 0.7 \wedge worseThan(F_i, H_T, H') > 0 \tag{6}$$

$$worseThan(F_i, H_T, H') = \frac{\sum_{H_c \in H'} 1[sent(F_i, H_T) < sent(F_i, H_c)]}{|H'|} \tag{7}$$

Then, we can construct a basic explanation as a set of pros and a set of cons as in Eqs. 8 and 9; for example, $Pros(H_T, H')$ is a set of tuples, each tuple comprising a pro feature and its $betterThan$ score and likewise for $Cons(H_T, H')$

$$Pros(H_T, H') = \{(F, v) : pro(F, H_T, H') \wedge v = betterThan(F, H_T, H')\} \tag{8}$$

$$Cons(H_T, H') = \{(F, v) : con(F, H_T, H') \wedge v = worseThan(F, H_T, H')\} \tag{9}$$

4.2 Personalised Explanations

The approach described in Sect. 4.1 treats each hotel feature equally, but in reality different features will matter to different users. If we wish to create compelling explanations then we will need to focus on those features that matter to the target user. For this, we assume we have access to user profiles made up of the same type of features as cases, each with a relative importance value to reflect the importance (*imp*) of the feature to the user as in Eq. 10. A more detailed account of the user profiling is beyond the scope of this work but briefly we create profiles just as we create hotel cases, as mentioned previously, by mining opinions from the user's reviews and mapping these review features to item features. Then we can calculate $imp(F_i, U)$ in a similar manner to how we calculated $imp(F_i, H)$: as the number of occurrences of F_i in $Reviews(U)$ divided by the total number of feature occurrences in $Reviews(U)$.

$$Profile(U) = \{[F_i, imp(F_i, U)] : F_i \in Reviews(U)\} \tag{10}$$

Now we can modify the way we generate the pros (or cons) of an explanation so that in addition to capturing the feature and its *betterThan* (or *worseThan*) scores we can also include an importance score for the target user U_T as in Eqs. 13 and 14.

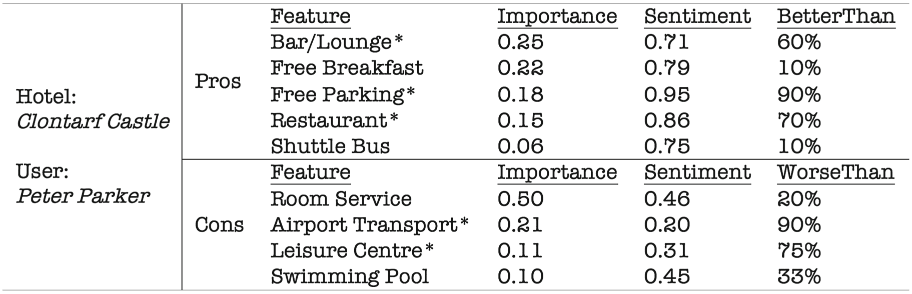
$$\begin{aligned} &pro(F,U_T, H_T, H') \leftrightarrow \\ &\qquad sent(F, H_T) > 0.7 \wedge betterThan(F, H_T, H') > 0 \wedge imp(F, U_T) > 0 \end{aligned} \tag{11}$$

$$\begin{aligned} &con(F,U_T, H_T, H') \leftrightarrow \\ &\qquad sent(F, H_T) < 0.7 \wedge worseThan(F, H_T, H') > 0 \wedge imp(F, U_T) > 0 \end{aligned} \tag{12}$$

$$\begin{aligned} &Pros(U_T, H_T, H') = \\ &\quad \{(F, v, m) : pro(F, U_T, H_T, H') \wedge v = betterThan(F, H_T, H') \wedge m = imp(F, U_T)\} \end{aligned} \tag{13}$$

$$\begin{aligned} &Cons(U_T, H_T, H') = \\ &\quad \{(F, v, m) : con(F, U_T, H_T, H') \wedge v = worseThan(F, H_T, H') \wedge m = imp(F, U_T)\} \end{aligned} \tag{14}$$

In this way, for a target user U_T and hotel H_T, as well as a set of alternative hotels H', we can construct an explanation for H_T relative to H' that emphasises those pros and cons that matter to U_T. An example explanation structure is shown in Fig. 2, for a user *Peter Parker* and a *Clontarf Castle Hotel* in Dublin. Based on the user's profile we can see that he is interested in a number of listed features including *Bar/Lounge, Free Breakfast, Airport Transport, Restaurant, Leisure Centre, Shuttle Bus, Swimming Pool, and Room Service*, in order of decreasing importance score. In *Clontarf Castle* some of these features have been positively reviewed in the past (high sentiment scores) and so are



		Feature	Importance	Sentiment	BetterThan
Hotel: *Clontarf Castle* User: *Peter Parker*	Pros	Bar/Lounge*	0.25	0.71	60%
		Free Breakfast	0.22	0.79	10%
		Free Parking*	0.18	0.95	90%
		Restaurant*	0.15	0.86	70%
		Shuttle Bus	0.06	0.75	10%
	Cons	Feature	Importance	Sentiment	WorseThan
		Room Service	0.50	0.46	20%
		Airport Transport*	0.21	0.20	90%
		Leisure Centre*	0.11	0.31	75%
		Swimming Pool	0.10	0.45	33%

Fig. 2. An example of a raw explanation structure showing pros and cons that matter to the user along with associated importance, sentiment, and better/worse than scores.

listed as pros (e.g. *Bar/Lounge and Restaurant*) while others have been more negatively reviewed (e.g. *Airport Transport and Swimming Pool*) and are listed as cons. In each case we can see the proportion of alternative recommendations that this hotel is better or worse than with respect to a particular pro or con, respectively. For example, *Clontarf Castle* has been reviewed very favourably for its *Free Parking* (sentiment of 0.95) and it is better for this than 90 % of the alternative recommendations. In contrast its *Leisure Centre* appears to be lacking (sentiment of only 0.31) and it is worse than 75 % of the alternatives. Of course there are also some features that matter to the user but that do not appear in the hotel's reviews and so these are not in the explanation.

4.3 Compelling Explanations

The explanation structure so far can be made up of a large number of features. In fact, as we shall see later, in our TripAdvisor dataset basic explanations tend to include an average of 6–7 pros and 2 or 3 cons. That is a lot of features to present to the user especially since not all of them will be very compelling. Many of the pros might be better than only a small fraction of the other recommendations. One option is to filter features based on how strong a reason they may be to choose or reject the target hotel case. We define a *compelling* feature to be one that has a $betterThan$ (pro) or $worseThan$ (con) score of $> 50\,\%$ instead of just > 0. Thus, a compelling pro is one that is better than a majority of alternative recommendations and a compelling con is one that is worse than a majority of alternatives. A compelling pro may be a strong reason to choose the target hotel; a compelling cons is a strong reason to avoid it.

We define a *compelling explanation* as a non-empty explanation which contains only compelling pros and/or compelling cons. For instance, referring back to Fig. 2, we have marked compelling features with an asterisk after their name; so, the compelling explanation derived from this basic explanation would include *Bar/Lounge, Free Parking, Restaurant* as pros and *Airport Transport* and *Leisure Centre* as cons. These are all features that matter to the user and they distinguish the hotel as either better or worse than a majority of alternatives.

4.4 Using Explanations to Rank Recommendations

A unique element of this work is our proposal to use explanations to rank recommendations. To do this we need to score explanations to reflect how strongly they are likely to be when convincing the user to choose (or reject) a given hotel; hotels with the strongest explanations should appear at the top of the ranking. To do this we use a straightforward scoring function to measure the strength of an explanation as the weighted sum of its pros minus the weighted sum of its cons as shown in Eq. 15.

$$\begin{aligned} strength(U_T, H_T, H') = & \\ & \sum_{f \in Pros(U_T, H_T, H')} betterThan(f, H_T, H') \times imp(f, U_T) - \\ & \sum_{f \in Cons(U_T, H_T, H')} worseThan(f, H_T, H') \times imp(f, U_T) \end{aligned} \tag{15}$$

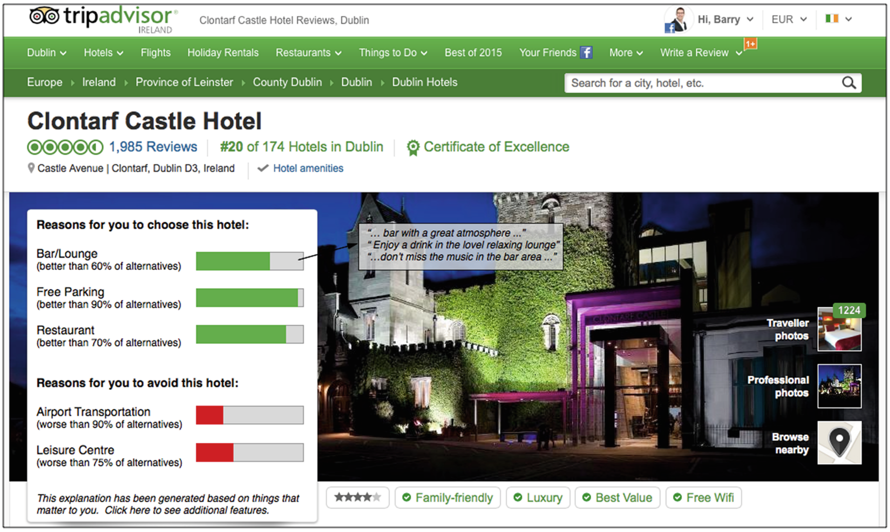
We can consider two versions of this scoring function, one that is applied to basic recommendations and one that is applied to compelling explanations. In each case the core calculation remains the same but only the features change. For example, applying the metric to the compelling features in Fig. 2 we calculate score of 0.15 based on a pro-score of 0.42 and a cons score of 0.27 (that is, $0.42 - 0.27 = 0.15$). Using this scoring function we can now rank-order hotels for recommendation in descending order of explanation strength.

4.5 Presenting Explanations to the User

So far we have said nothing about how these explanations might be presented to the user. For completeness, in Fig. 3 we illustrate one example for Clontarf Castle. The explanation is in the pop-up on the main hotel photo. We show the compelling version of the explanation with 3 pros and 2 cons.

The pros and cons are ordered based on their importance to the target user. The horizontal (sentiment) bar next to each shows the relative sentiment associated with the feature and beneath each is an indication of the *betterThan* or *worseThan* score, as appropriate. Evidently, Clontarf Castle is superior to a significant majority of alternatives in terms of its *Bar/Lounge*, *Free Parking*, and *Restaurant*, all of which are important to the target user, but it loses out to a majority of alternatives in terms of its *Airport Transportation* and *Leisure Centre*.

The user can request a more detailed explanation to reveal the full set of explanation features. By hovering over a sentiment bar the user can see a summary of the opinions extracted from reviews about that feature; this is shown for the *Bar/Lounge* feature in Fig. 3. And by clicking on the text that references alternatives the user will be brought to a list of the relevant alternatives;



Clontarf Castle Hotel

1,985 Reviews | #20 of 174 Hotels in Dublin | Certificate of Excellence

Castle Avenue | Clontarf, Dublin D3, Ireland | ✓ Hotel amenities

Reasons for you to choose this hotel:

Bar/Lounge (better than 60% of alternatives)	🟩
Free Parking (better than 90% of alternatives)	🟩
Restaurant (better than 70% of alternatives)	🟩

Reasons for you to avoid this hotel:

Airport Transportation (worse than 90% of alternatives)	🟥
Leisure Centre (worse than 75% of alternatives)	🟥

This explanation has been generated based on things that matter to you. Click here to see additional features.

"... bar with a great atmosphere ..."
"Enjoy a drink in the lovel relaxing lounge"
"...don't miss the music in the bar area ..."

Fig. 3. An example explanation showing pros and cons that matter to the target user along with sentiment indicators (horizontal bars) and information about how this item fares with respect to alternatives.

for example, if the user selected the "worse than 75 % of alternatives" for the Leisure Centre feature she would be brought to a list of these superior alternatives. In this way explanations also serve as a navigation structure to help users navigate between these very alternatives. This is just one approach to presenting explanations to the user and future work will consider interface issues further.

5 Evaluation

There are 4 important aspects to our approach to generating opinionated explanations for recommendation: (i) we separately emphasise the pros and cons of each item; (ii) we use information about features that matter to the user to personalise these explanations; (iii) we link the explanation to recommendations which offer better or worse feature options; (iv) we propose to use these explanation structures for the ranking of recommendations themselves. In combination we believe that these aspects make for a novel and potentially powerful approach to explanations for recommender systems and we provide some evaluation data to support this in what follows.

5.1 Data and Methodology

We use a TripAdvisor dataset as a source of users, reviews, and hotels. This dataset contains 1,000 users who have each written at least 10 hotel reviews for 2,370 hotels that they had booked. These reviews are used for user profiles. In addition we had more than 220,000 reviews by almost 150,000 reviewers available for the hotel cases.

For each target user U_T we select a hotel that they have booked, H_B, and collect a set of 10 related hotels from TripAdvisor. These additional hotels are those that TripAdvisor recommends as *related hotels*; we understand that TripAdvisor generates these using a combination of location, similar users, and meta-data.

Our intention is to simulate a typical session in which U_T has located a hotel of interest H_B, and a set of alternatives suggested by TripAdvisor. The booked hotel and the alternatives represent a set of recommendations for U_T. For each such session we generate an explanation for each of the 11 recommended hotels for U_T; in fact we will generate a basic explanation and a compelling explanation for each hotel. We analyse various properties of these explanations in addition to their utility for ranking the hotels for recommendation.

5.2 Pros vs. Cons, Better vs. Worse

First we investigate the number of pros and cons and their $betterThan$/ $worseThan$ scores. Starting with basic explanations, Fig. 4(a) shows the average number of pros and cons generated per explanation (left y-axis) and also the average $betterThan/worseThan$ scores (right y-axis). We can see that on average we are recommending about 5.8 pros versus only 2.2 cons reflecting the strong positive bias amongst reviews.

Interestingly we see a significant difference between the average $betterThan$ score for pros (0.42) compared to the average $worseThan$ score for cons (0.63). In other words, for a typical hotel, its pros will typically be better than about 42 % of the alternatives in the recommendation session. In contrast, when it comes to the cons, it is usually the case that the hotel in question does worse than most of the alternatives in the recommendation session.

Figure 4(b) shows corresponding results for compelling explanations. Incidentally about 97 % of the basic explanations are compelling. Now we can see that the average number of pros and cons is more balanced; there are 1.76 pros vs 1.55 cons. The average $betterThan$ and $worseThan$ scores for these explanations are 70 % and 75 %, respectively. These explanations are simpler to interpret

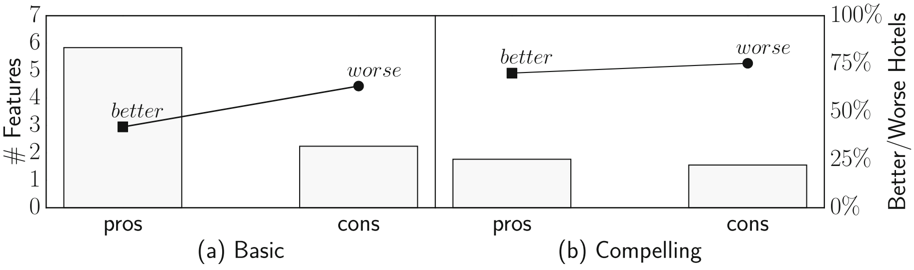


Fig. 4. The average number of pros and cons and the average $betterThan$ and $worseThan$ scores per explanation for basic explanations and compelling explanations.

(having fewer features) and more compelling in the sense that their features are better/worse that a strong majority of alternatives. Intuitively this combination of simplicity and compellingness should make them effective when it comes to helping users to decide, be it to accept or reject a given recommendation.

5.3 Using Explanations to Rank Recommendations

Earlier we described how to compute the *strength* of an explanation as a function of its pros and cons (see Eq. 15) and we proposed to use this score to rank hotels for recommendations. To evaluate how well this might work we need a ground-truth against which to judge our hotel recommendations. We propose the average rating that is available alongside each TripAdvisor hotel for this; a similar approach has been used by [6,7].

For each recommendation session we re-rank the recommended hotels (including the booked hotel) according to the strength of their basic and compelling recommendations and note the average position of the booked hotel. Then we compare the average rating of the booked hotel to the average ratings of the hotels above and below the booked hotel in the ranking. Ideally we would like to see all hotels above the booked hotel to have better average ratings and all hotels ranked below the booked hotel to have lower average ratings.

It noteworthy that we can expect this to be a tough test. After all the user chose the booked hotel for a reason and so we can expect it to be a highly rated one, all things being equal. Related to this, it is also worth noting that the average ratings for hotels in the recommendation sessions tend to be very high — TripAdvisor is unlikely to suggest poorly rated hotels — and so rating diversity can be low within sessions providing little opportunity for measurable ranking improvement. To deal with this we ordered our sessions based on variance of average user ratings (across the hotels in each session) and selected the top 20 % (200 sessions) that had the highest average user rating variance.

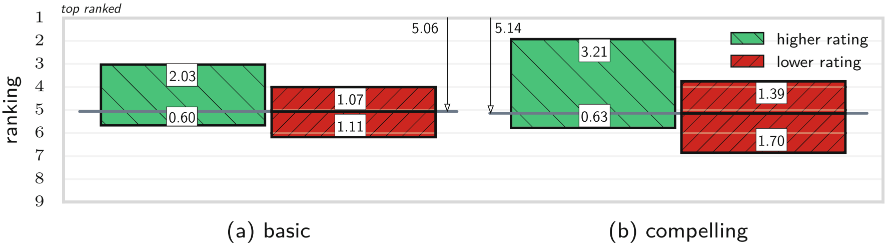


Fig. 5. A comparison of recommendation ranking results for basic and compelling explanations, showing only the first 9 rank positions. The solid horizontal lines indicate the average rank of the booked hotel. In each pair of bars, the green (left) show the average number of better-rated hotels above and below the booked hotel using our rankings. And in each pair the red bars (right) show the number of lower rated hotels above and below the booked hotel.

The results are shown in Fig. 5 as a bar chart that needs some explanation. First the horizontal lines that bound the charts at the top and the bottom represent the position of the top ranked hotel (position 1 in the ranking) and the position of the bottom ranked hotel (position 10 in the ranking). Between these boundaries there are two separate bar charts for the rankings based on the strength of: (a) basic explanations; (b) compelling explanations. The horizontal origin-line for each bar-chart is positioned between the top and bottom boundaries to reflect the average position of the booked hotel in each session. For basic explanations the booked hotel is ranked on average at position 5.06 and for compelling explanations the booked hotel is ranked a little lower at 5.14.

Next, each bar-chart contains 2 bars to reflect the number of recommendations that have a higher average rating than the booked hotel (the left-hand bar) and the number of recommendations with a lower average user rating than the booked hotel (the right-hand bar). The vertical position of these bars relative to the origin-line indicates whether these higher or lower rated hotels appear above or below the booked hotel in the ranking.

For example, for the compelling explanations (Fig. 5(b)) we see: the booked hotel ranked at position 5.14; an average of 3.21 recommendations above it with higher average ratings; only 0.63 hotels with higher ratings ranked below it (left-hand bar). This is good because it means that by ranking hotels by the strength of their explanations we are able to produce a ranking that tends to push a large majority (84 %) of higher rated hotels above the booked hotel. Next we look at the bar corresponding to lower rated hotels (right hand bar). Most of these lower rated hotels (1.70) are ranked below the booked hotel, but some (1.39) are ranked above. Again this is positive as it means that our explanation-based ranking tends to rank most of the poorer quality hotels below the booked one, although sometimes a lower rated hotel is ranked above the booked hotel. The results are broadly similar when we look at the basic explanations, although slightly fewer higher ranked hotels appear above the booked hotel.

6 Discussion and Limitations

To sum up, we have described a novel approach to generating explanations for opinionated recommender systems that can be used not only to help justify recommendations to users, but also to influence the recommendation ranking. In the space available we have left out many details and a number of items remain open for discussion, for example:

1. In our evaluation we base our profiles on reviews that have been authored by users but this introduces a significant cold-start problem in practice since most users are not active reviewers. Nevertheless there are many other ways to generate profiles such as mining opinions from reviews that users have rated, liked, or simply read. Moreover, even when user profiles remain lacking in features we could use those features we have to identify *similar users* and harness their profiles (and the features that matter to them) when generating explanations for the target user.

2. We have also said relatively little about how explanations might be presented to users, other than by showing one concrete example. Again this is a matter for future work where we will consider a variety of recommendation interfaces and styles, each emphasising different aspects of explanations. It will be interesting to see which styles users will find most helpful and compelling, and whether these do in fact support more satisfactory choices.
3. While the evaluation results on ranking are far from conclusive, they do suggest that using explanations for ranking can deliver a high quality ordering of recommendations. Indeed, when we compute the average rank correlation between the ground-truth (average TripAdvisor rating) ordering and the basic or compelling based orderings, we find correlation values of approximately 0.62 indicating a reasonable correlation between our explanation-based rankings and the ground-truth; this is yet another sign that the explanation-based approach is effective for recommendation ranking.
4. Finally, we have limited our research, thus far, to focusing on hotel reviews from TripAdvsor. However, there is nothing in the work that suggests this should be a limitation. In fact earlier work by [6–8] has applied similar opinion mining techniques to good effect to other types of user reviews such as those found on Amazon for consumer electronic products.

7 Conclusions

This work builds on recent research in the case-based reasoning community by bringing together ideas from CBR, opinion mining, and recommender systems. Its main contribution is a novel approach to explanation that can also be used to influence recommendation ranking. Rather than relying on similarity as a proxy for user relevance we base recommendation decisions on the ability to explain/justify recommendations to the user; this bears a resemblance to the work [22] which proposed the use of adaptation knowledge as a part of the case retrieval and ranking process, arguing that *adaptability* served as a more reliable metric for retrieval than traditional notions of similarity.

This is very much a work in progress. We have described our approach to generating explanations and provided some point examples on how such explanations might be used in practice. We analysed the structure of these explanations based on a TripAdvisor dataset of hotel reviews. We demonstrated that it is feasible to generate compelling explanations as part of the recommendation process, and that these explanations could be used for effective ranking.

Future work will focus on live-user trials of this approach. This will include experimenting with different presentation formats for our explanation structures to investigate whether users find them more or less useful, and whether there is evidence to suggest that such explanations do lead to better decisions in practice.

Acknowledgments. This work is supported by the Insight Centre for Data Analytics under grant number SFI/12/RC/2289.

References

1. Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.: Explaining collaborative filtering recommendations. In: Proceedings of The ACM Conference on Computer Supported Cooperative Work, pp. 241–250, ACM (2000)
2. Symeonidis, P., Nanopoulos, A., Manolopoulos, Y.: Providing justifications in recommender systems. IEEE Trans. Syst. Man Cybern. Part A Syst. Hum. **38**(6), 1262–1272 (2008)
3. Pu, P., Chen, L.: Trust-Inspiring explanation interfaces for recommender systems. Knowl. Based Syst. **20**(6), 542–556 (2007)
4. Coyle, M., Smyth, B.: Explaining search results. In: Proceedings of The 19th International Joint Conference on Artificial Intelligence, pp. 1553–1555, Morgan Kaufmann Publishers Inc (2005)
5. Friedrich, G., Zanker, M.: A taxonomy for generating explanations in recommender systems. AI Mag. **32**(3), 90–98 (2011)
6. Dong, R., Schaal, M., O'Mahony, M.P., McCarthy, K., Smyth, B.: Mining features and sentiment from review experiences. In: Delany, S.J., Ontañón, S. (eds.) ICCBR 2013. LNCS, vol. 7969, pp. 59–73. Springer, Heidelberg (2013)
7. Dong, R., O'Mahony, M.P., Smyth, B.: Further experiments in opinionated product recommendation. In: Lamontagne, L., Plaza, E. (eds.) ICCBR 2014. LNCS, vol. 8765, pp. 110–124. Springer, Heidelberg (2014)
8. Dong, R., Schaal, M., O'Mahony, M.P., Smyth, B.: Topic extraction from online reviews for classification and recommendation. In: Proceedings of The 23rd International Joint Conference on Artificial Intelligence (2013)
9. Buchanan, B.G., Shortliffe, E.H.: Rule Based Expert Systems: The Mycin Experiments of the Stanford Heuristic Programming Project. The Addison-Wesley Series in Artificial Intelligence, vol. 3. Addison-Wesley Longman Publishing Co., Inc., Boston (1984)
10. Sørmo, F., Cassens, J., Aamodt, A.: Explanation in case based reasoning perspectives and goals. Artif. Intell. Rev. **24**(2), 109–143 (2005)
11. McSherry, D.: Explaining the pros and cons of conclusions in CBR. In: Funk, P., González Calero, P.A. (eds.) ECCBR 2004. LNCS (LNAI), vol. 3155, pp. 317–330. Springer, Heidelberg (2004)
12. Doyle, D., Cunningham, P., Bridge, D.G., Rahman, Y.: Explanation oriented retrieval. In: Funk, P., González Calero, P.A. (eds.) ECCBR 2004. LNCS (LNAI), vol. 3155, pp. 157–168. Springer, Heidelberg (2004)
13. Druzdzel, M.J.: Qualitative verbal explanations in Bayesian belief networks. Artif. Intell. Simul. Behav. Q. Spec. Issue Bayesian Netw. **94**, 43–54 (1996)
14. Bilgic, M., Mooney, R.J.: Explaining recommendations: Satisfaction vs. Promotion. In: Proceedings of Beyond Personalization 2005: A Workshop on the Next Stage of Recommender Systems Research at The 2005 International Conference on Intelligent User Interfaces, pp. 13–18 (2005)
15. Vig, J., Sen, S., Riedl, J.: Tagsplanations: explaining recommendations using tags. In: Proceedings of The 13th International Conference on Intelligent User Interfaces, pp. 47–56, ACM Press (2008)
16. Reilly, J., McCarthy, K., McGinty, L., Smyth, B.: Explaining compound critiques. Artif. Intell. Rev. **24**(2), 199–220 (2005)
17. Tintarev, N., Masthoff, J.: The effectiveness of personalized movie explanations: an experiment using commercial meta-data. In: Nejdl, W., Kay, J., Pu, P., Herder, E. (eds.) AH 2008. LNCS, vol. 5149, pp. 204–213. Springer, Heidelberg (2008)

18. McSherry, D.: Similarity and compromise. In: Ashley, K.D., Bridge, D.G. (eds.) ICCBR 2003. LNCS, vol. 2689, pp. 291–305. Springer, Heidelberg (2003)
19. Hu, M., Liu, B.: Mining opinion features in customer reviews. In: Proceedings of The 19th National Conference on Artificial Intelligence, pp. 755–760, AAAI Press (2004)
20. Justeson, J.S., Katz, S.M.: Technical terminology: some linguistic properties and an algorithm for identification in text. Nat. Lang. Eng. **1**(1), 9–27 (1995)
21. Moghaddam, S., Ester, M.: Opinion digger: an unsupervised opinion miner from unstructured product reviews. In: Proceedings of The 19th ACM International Conference on Information and Knowledge Management, pp. 1825–1828, ACM Press (2010)
22. Smyth, B., Keane, M.: Adaptation-Guided retrieval: questioning the similarity assumption in reasoning. Artif. Intell. **102**(2), 249–293 (1998)