Accuracy in Rating and Recommending Item Features

Lloyd Rutledge[1⋆], Natalia Stash[2], Yiwen Wang[2], and Lora Aroyo[3]

[1] Telematica Instituut, Enschede, The Netherlands
[2] Technische Universiteit Eindhoven, Eindhoven, The Netherlands
[3] Vrije Universiteit, Amsterdam, The Netherlands

Abstract. This paper discusses accuracy in processing ratings of and recommendations for item features. Such processing facilitates feature-based user navigation in recommender system interfaces. Item features, often in the form of tags, categories or meta-data, are becoming important hypertext components of recommender interfaces. Recommending features would help unfamiliar users navigate in such environments. This work explores techniques for improving feature recommendation accuracy. Conversely, it also examines possibilities for processing user ratings of features to improve recommendation of both features and items.
This work's illustrative implementation is a web portal for a museum collection that lets users browse, rate and receive recommendations for both artworks and interrelated topics about them. Accuracy measurements compare proposed techniques for processing feature ratings and recommending features. Resulting techniques recommend features with relative accuracy. Analysis indicates that processing ratings of either features or items does not improve accuracy of recommending the other.

1 Introduction

Recommender systems have acquired an important role in guiding users to items that interest them. Traditionally, recommendation systems work exclusively with tangible objects (such as films [5], books or purchasable products [8]) as what they let users rate and what they consequently recommend. More recently, however, abstract concepts related to such items play an increasingly important role in extended hypertext environments around recommender systems. For example, Amazon.com's recommender system[1] lets users select categories to fine-tune recommendation lists. In addition, Amazon.com lets users assign tags to items, which extends not only search for and navigation between items but also recommendation of them. As tags, categories and other concepts become more important to users in interaction with recommender systems, users will benefit from help with finding appropriate ones. The context of recommender systems offers an obvious tool for this: the rating and recommendation of such concepts.

⋆ Lloyd Rutledge is also affiliated with CWI and the Open Universiteit Nederland
[1] `http://www.amazon.com/gp/yourstore/`

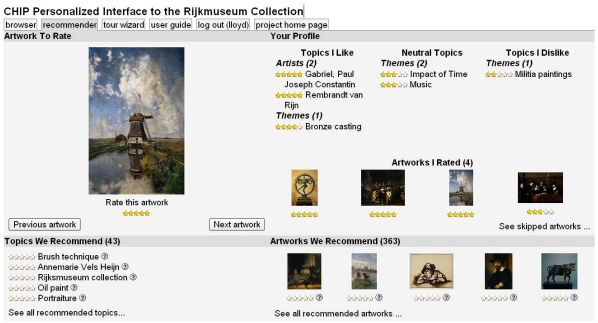

Fig. 1. CHIP Artwork Recommender display

Such rating and recommending of abstract concepts occurs in the CHIP project Artwork Recommender[2] [1]. Figure 1 shows an example display. The system's users can rate and recommend items in the form of artworks from the collection of the Rijksmuseum Amsterdam. Users can also rate and recommend features in the form of abstract topics (such as artist, material and technique) related to these artworks, which fall in a hyperlinked network joining artworks with related topics and topics with each other. Recommending artworks and topics brings users to interface displays from which they can rate related artworks and topics, improving their profiles. In this process, users not only find artworks they like, they learn about personally interesting art history topics that affect their taste. Studies show that users benefit from feature recommendation in such an integrated environment [11].

This paper explores how to maximize both the accuracy of feature recommendation and the exploitation of feature ratings. It starts by discussing related work and describing the evaluation methods it applies. The first core section discusses the differences between how users rate features and how they rate items. The paper then shows the impact on collaborative filtering accuracy that feature rating and recommending bring. The final core section proposes an adaptation of established content-based techniques to recommend features. This paper wraps up with conclusions from the study.

2 Related Work

This section discusses work related to navigating and rating features in recommender systems. This work tends to fall in the separate subfields of feature-based navigation in recommender systems, rating of tags and browsers for extensively annotated items. These fields combine in the implementation this work applies in exploring recommender accuracy for these topics.

Amazon.com uses both categories and tags in its recommender service by letting users specify that each can refine recommendation lists. Amazon, however,

[2] `http://www.chip-project.org/demo`

lets users rate neither categories nor tags, only items. In addition, they do not use categories and tags in recommendation generation processing. MovieLens recently added tags to its recommender service, giving users both the ability to assign tags to movies and to rate tags assigned by others [10]. This rating of tags relates closely to this work's rating of features. In MovieLens, however, a user's rating of a tag indicates the user's confidence in its informative accuracy rather than how appealing the user finds that topic.

While Amazon.com and MovieLens only let users rate items, Revyu.com lets users rate "anything" by assigning ratings (and descriptive reviews) to community-defined tags [6]. These ratings represent level of user interest. Revyu.com does not distinguish between items and their features because tags can represent either in the same manner. However, Revyu.com does not process these ratings for recommendations.

Amazon.com and MovieLens offer tags as part of recommendation, providing community-defined features. Amazon.com also provides centrally maintained item features in the form of categories. Facetted browsers offer the current state-of-the-art for accessing items by exploiting their centrally maintained features, where the features are more complex in nature. Typically, with facetted browsers, items can have many features and each feature is a property assignment using one of multiple property types. The E-Culture browser[3] offers such facetted access for museum artworks, processing data for over 7000 artworks from multiple institutions that cooperatively apply common vocabularies in making typed properties of these artworks [9]. The annotations from the CHIP Artwork Recommender use the same vocabularies and property types, which has enabled incorporation of the Rijksmuseum artworks and annotations into the E-Culture browser.

Studies with the CHIP Artwork Recommender show that coordinated rating and recommending of features with items improves how novices learn art topics of interest [11]. Other studies with this system show that explaining item recommendation in terms of common features is important for user assessment of recommender system competence and other aspects of trust [4]. This work now performs similar accuracy analyses for feature recommendation and for processing feature ratings for recommendation in general.

3 Method

This section presents the methods that evaluate the techniques this work proposes. It first discusses the user tasks to which the evaluating measurements apply. It then describes the application of the leave-n out approach that provides accuracy measurements here. This section wraps up by presenting the specific metrics for measuring the satisfaction of these user tasks.

The CHIP recommender interface display in Figure 1 illustrates several user tasks. This work's evaluation focuses on two of these tasks. Both involve providing recommendations as a list of all things the user is likely to like. One task

[3] `http://e-culture.multimedian.nl/demo/search`

is showing all recommendations of items, to which the interface provides access from the link "See all recommended artworks" at the bottom right of Figure 1. The area above this links shows the top five of these recommendations. The second task is show all recommended features, which the link "See all recommended topics" links to at the bottom left of Figure 1.

This work's evaluations of the techniques it proposes apply the leave-n out approach. This starts by withholding 10% of the sample ratings as a truth set. The algorithms to evaluate then process the remaining ratings to calculate predictions for the ratings in this truth set. Comparing the predictions with their corresponding true values forms the basis for the various metrics these evaluations use.

The metrics that this work calculates in its evaluations are NMAE, precision and recall. They are common in recommender system and information retrieval research. As both main user tasks involve retrieval of all appropriate matches, these classic metrics of precision and recall apply well.

The *NMAE* (normalized mean absolute error) measures predictive accuracy by showing by what percentage the system's predictions for the truth set ratings differ from their real values. The remaining metrics provide classification accuracy, which measures how well the system generates list of recommendations. *Precision* shows how many of the recommendations the user truly likes. *Recall* indicates how many desired items and features appear among the recommendations. Precision and recall depend on a recommendation threshold, which is a value above which predicted ratings form recommendations for their corresponding concepts. That is, the system recommends an item or feature if the predicted interested for it exceeds this threshold.

4 User Ratings for Items and Features

This section discusses patterns that emerge in comparing how users rate items with how they rate features. This analysis uses two sets of ratings for artworks and related art topics entered by users of the CHIP Artwork Recommender. One set of ratings comes from the online demo, with no restriction on use. The other is from a directed user study. By having ratings for both artworks and topics, this set represents ratings for items and features respectively.

Figure 2 shows the distribution of these ratings across the users. The sample sets for this current work includes only users who gave at least 10 ratings, of which at least one is for a feature, in order to ensure there is substantial data from each user from which to calculate recommendations. The bar charts on the right half of Figure 2 show overall a large amount of five-star (value is 1) and four-star (value is 0.5) ratings.

One rating set comes from the main online demo for the CHIP Artwork Recommender. These users have *"free use"* of the online demo in the sense that they are unsupervised and can have as many sessions as they want whenever they want with no particular tasks to fulfill and no restrictions in how to use the demo. This usage represents the general target use of a recommender system.

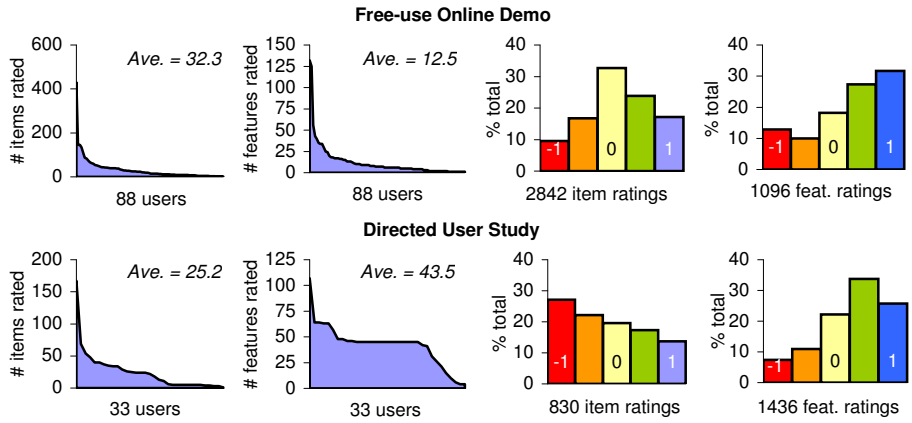

Fig. 2. User-rating distributions

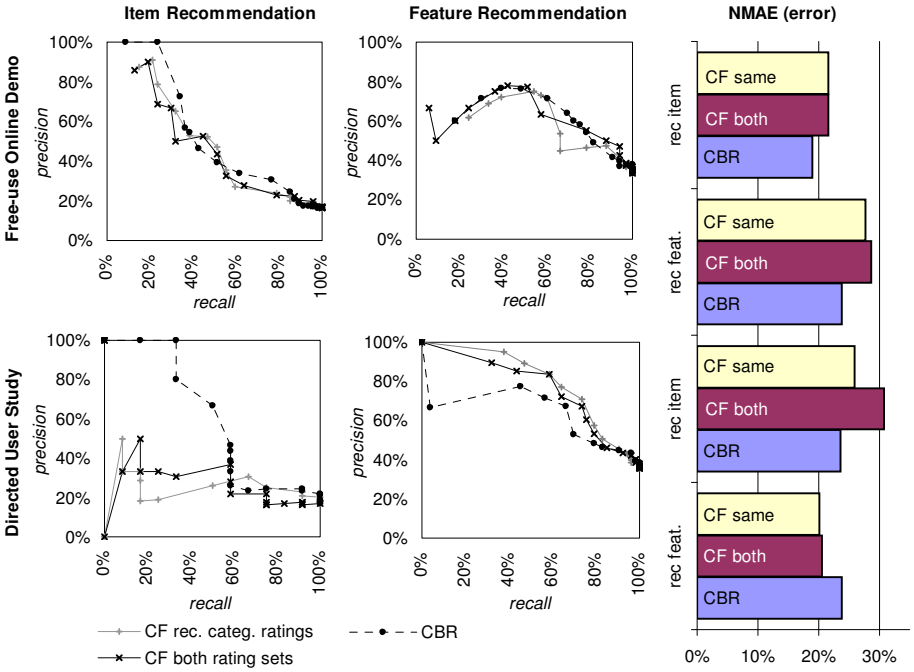

Fig. 3. Accuracy charts for proposed techniques

One pattern that Figure 2 shows is that users given free use of such a system tend to enter many more, in this case almost three times as many, ratings for items as they do for features. Another pattern is that feature ratings tend to be more positive and extreme than item ratings. Users were almost twice as likely to rate a feature with five stars (value is 1) than an item. They were also almost twice as likely to rate an item as neutral (value is 0) than a feature. One possible explanation is that users have more extreme opinions about features than items because features are abstract generalizations whereas an item can have many potentially contrasting characteristics that affect user interest in it.

Another possible explanation for the more frequently positive feature ratings is that previous familiarity has a different impact on rating items than on rating features. Perhaps users are more familiar with topics that influence their interest, particularly if this influence is positive. Because users see images of items, they can quickly make a rating for any item, even if they have not seen it before. Features, on the other hand, appear as text labels instead of images, meaning that users must be previously familiar with a topic to enter a rating for it.

While the previous rating set comes from free use of the online demo, another sample set comes from a *directed user study* of this system. This study starts by showing its users 45 topics, which this work considers features, and asking the user to rate them. It then has the user interact with the main demo for a minimum amount of time.

The directed user study brought different patterns in the charts in Figure 2 than for the free-use online demo. The two left-most charts, with distributions of each type of rating across the users, are flatter than for the free-use demo. One factor is that, in the directed study, ratings for each user came from a single session with a time duration minimum. The plateau in the curve for the distribution of feature ratings across users reflects the 45 features the study asks users to rate. The values for the ratings also spread more evenly for the directed study than for the free-use demo. This may be because the directed study compels users to rate a particular variety of features and, to a lesser degree, items. As with the free-use demo, users of the directed study tend to give more positive ratings to features than to items, although the directed study's feature rating values tend to be more moderate.

5 Collaborative Filtering

Collaborative filtering (CF) is the determination of similarity patterns in ratings from multiple users in order to recommend to a user what similar users rate highly. This is typically the processing of item ratings to recommend items. The software for CF that this work extends is the open source Duine toolkit for recommender system frameworks[4], which applies the Pearson correlation coefficient [7]. This section explores the impact on CF of both the rating and recommendation of item features, showing that CF provides accurate feature

[4] `http://sourceforge.net/projects/duine`

recommendation but does not improve accuracy when processing ratings from both features and items together.

Figure 3 indicates the accuracy of the proposed techniques. It plots the corresponding precision and recall values from 21 thresholds evenly spread in the full range of rating values. The threshold for recommendation that the precision and recall measurements here use is the top 20% of the range. The bar charts along the right show predictive accuracy for these techniques. The top half comes from the free-use set, while the bottom half comes from the directed user study. In each case, the NMAE charts show measurements for recommending items and features separately. Here, the bar "CBR" indicates the accuracy of content-based recommendation, which the next section discusses. "CF both" measures the error resulting from processing ratings of items and features together with collaborative filtering. Finally, "CF same" processes only item ratings for recommending items and feature ratings for recommending features.

The charts in Figure 3 show that, given this work's rating sets, CF works as well for features as for items. This indicates that systems can recommend features with comparable accuracy as they do for items. One indication of this comparable performance is in predictive accuracy, which the NMAE bar charts on the right of Figure 3 show. Here, CF predictive accuracy from the larger feature rating set has the same average error, if not slightly less, than CF recommendation from the larger item rating set. This comparison uses the largest available set for each recommendation category because CF relies on large amounts of ratings. As Figure 2 shows, the larger set of feature ratings comes from the directed study, which Figure 3's bottom-most bar triple conveys. The larger set of item ratings comes from the free-use demo, which the top-most bar trio conveys.

Another indication of CF's accuracy in feature recommendation comes from the precision-recall plot graphs in Figure 3. They convey that CF provides better classification accuracy for recommending features than items. As with the predictive accuracy comparison, this comparison is between CF for the larger ratings sets: the predictive accuracy for the directed study's feature ratings, which the lower right plot graph in Figure 3 shows, with the predictive accuracy for the feature ratings from the online demo in the upper left plot graph. The curve for features is clearly higher, with the points for each threshold having higher precision and recall than the corresponding points in the plot graph for items. While it is tempting to conclude from this that features in general result in such strongly more accurate CF classification than items, a primary factor in the better classification in this comparison may be the strong overlap in ratings between users for the directed study's set of 45 topics. However, even if this overlap screws the comparison, the conclusion would still be that CF provides accurate classification at least when it compels users to rate overlapping sets.

While this work shows that accurate feature CF is possible, it fails to show how feature ratings can benefit item CF, and vice versa. Figure 3's NMAE bar charts shows that combining ratings sets in CF provides less predictive accuracy than CF processing of only ratings for the type of recommended concept. Here, the "CF same" bars show accuracy for CF processing of ratings for the type

of recommendation, either for items or for features. The "CF both" bars show accuracy for processing both item and feature ratings together and equivalently for each type of recommendation. Although processing both ratings sets provides more information than either alone, in all four cases there is either no discernable change or substantial decrease in accuracy in CF for the combined rating sets.

Figure 3's classification plot graphs show the equivalent degradation in accuracy for CF with combined rating sets. In three of the four graphs, the curve for combined processing is clearly lower than the other CF curve. The exception is feature CF with the free-use ratings, which show slight increase in precision but larger decrease in recall. That CF accuracy for both prediction and classification mostly decreases indicates that CF system should use only item ratings for item recommendation and only feature ratings for feature recommendation.

One possibility for having CF improved accuracy by combine rating sets is to treat items and features as domains in cross-domain mediation [3]. This approach shows that CF in one domain can improve with ratings from another by first computing user similarity in each domain separately, combining them and then applying the result for recommendation in the current domain. It remains an open challenge for cross-domain mediation or other techniques to exploit user ratings for either items or features to improve recommendation of the other.

6 Role-reversed Content-based Recommendation

Content-based recommendation (CBR) is the typical companion recommender algorithm to CF. While CF uses similarities between users in terms of ratings in order to recommend items similar users like, CBR uses similarities between items in terms of their features in order to recommend items similar to other items the current user likes. CF is typically more accurate than CBR when there are enough ratings for enough items from enough users. However, in the cold-start period leading up to this point, CBR typically performs better. Hybrid recommender systems provide best overall recommendation by selecting which of the two to apply in each recommendation situation [2]. With the previous section having established that CF can accurately recommend features, this section adapts CBR to do so as well, providing in combination the components needed for accurate hybrid recommendation of features. This adaptation is a "role-reversed CBR" to recommend features instead of items.

Core issues in CBR include assigning appropriate features for the items to recommend and determining appropriate processing for these features. This work uses a CBR technique for processing features that are item properties encoded with Semantic Web formats. The item and feature set are the museum artworks and annotations from the CHIP Artwork Recommender. This paper adapts established CBR algorithms in the following ways:

- treating semantically assigned properties as features (instead of keywords)
- assigning weights to features by adjusting tf-idf for frequency of properties
- processing cosine similarity on the resulting feature vectors

This technique represents the typical perspective of recommender systems, in which items to recommend are tangible objects, with features that are abstract concepts related to them. This section proposes *role-reversed CBR* for effective feature recommendation as CBR that switches the roles items and features play in its processing. That it, role-reversed processes features as "items" to recommend and applies cosine similarity with tf-idf weights on vectors consisting of the original items that each original feature annotates.

Figure 3 shows that this role-reversed CBR for features has similar precision and recall as CF for features. The curves in both feature recommendation plot graphs follow roughly the same pattern. For the free-use rating set, the curves are very close. For the directed study set, however, CBR precision tends to be less, albeit with roughly the same recall. A factor here may be that the second set has more ratings overall and many more ratings per user, conditions which typically improve CF in comparison to CBR. These measurements indicate that role-reversed CBR provides effective cold-start feature recommendation and can combine well with feature CF in hybrid systems for overall accurate feature recommendation. As with CF, while CBR can accurately recommend features, it remains an open challenge to have CBR exploit feature ratings to improve item recommendation, and vice versa.

7 Summary

This paper shows how to improve feature recommendation and what role processing ratings of either features or items has in recommending the other. Systems can recommend features with accuracy that is comparable to item recommendation. Techniques for doing so include CF and this work's role-reversed CBR. Users choose freely to rate features, although they rate items more frequently. Users tend to rate features more positively than items. It remains a challenge in both CF and CBR to have processing ratings for either items or features improve the recommendation of the either.

8 Acknowledgements

This work was a collaboration with the Rijksmuseum Amsterdam within the CHIP project[5] of the CATCH program[6], funded by the Dutch Organization for Scientific Research (NWO). The MultimediaN/E-Culture project[7] provides the encoding of the Getty vocabularies that define some of this work's items features, along with mappings to them from original curator annotations. The CATCH/STITCH project[8] provides an encoding for Iconclass, which defines other item features this work uses. The implementation here uses the open source

[5] http://www.chip-project.org/
[6] http://www.nwo.nl/catch
[7] http://e-culture.multimedian.nl/
[8] http://stitch.cs.vu.nl/

Duine Toolkit[9] for recommendation processing. The Rijksmuseum Amsterdam gave permission for the use of its images.

References

1. L. Aroyo, R. Brussee, L. Rutledge, P. Gorgels, N. Stash, and Y. Wang. Personalized museum experience: the Rijksmuseum use case. In *Museums and the Web 2007*, San Francisco, USA, April 11-14 2007.
2. M. Balabanovic and Y. Shoham. Fab: content-based, collaborative recommendation. *Commun. ACM*, 40(3):66–72, 1997.
3. S. Berkovsky, T. Kuflik, and F. Ricci. Cross-domain mediation in collaborative filtering. In *The 11th International Conference on User Modeling (UM2007)*, pages 355–359, June 2007.
4. H. Cramer, B. Wielinga, S. Ramlal, V. Evers, L. Rutledge, and N. Stash. The effects of transparency on perceived and actual competence of a content-based recommender. In *CHI 2008 Semantic Web User Interaction Workshop (SWUI 2008)*, Florence, Italy, April 5 2008.
5. N. Good, J. B. Schafer, J. A. Konstan, A. Borchers, B. M. Sarwar, J. L. Herlocker, and J. Riedl. Combining collaborative filtering with personal agents for better recommendations. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99)*, pages 439–446, Orlando, Florida, USA, July 18-22 1999.
6. T. Heath and E. Motta. Revyu.com: A reviewing and rating site for the web of data. In *The 6th International Semantic Web Conference (ISWC 2007)*, pages 895–902, Busan, Korea, November 2007.
7. J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl. An algorithmic framework for performing collaborative filtering. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 230–237, New York, NY, USA, 1999. ACM Press.
8. G. Linden, B. Smith, and J. York. Amazon.com recommendations: item-to-item collaborative filtering. *Internet Computing, IEEE*, 7(1):76–80, 2003.
9. G. Schreiber, A. Amin, M. van Assem, V. de Boer, L. Hardman, M. Hildebrand, L. Hollink, Z. Huang, J. van Kersen, M. de Niet, B. Omelayenko, J. van Ossenbruggen, R. Siebes, J. Taekema, J. Wielemaker, and B. Wielinga. MultimediaN E-Culture demonstrator. In *Proceedings of the Fifth International Semantic Web Conference (ISWC'06)*, pages 951–958, Athens, Georgia, USA, November 2006.
10. S. Sen, S. K. Lam, A. M. Rashid, D. Cosley, D. Frankowski, J. Osterhouse, F. M. Harper, and J. Riedl. tagging, communities, vocabulary, evolution. In *CSCW '06: Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*, pages 181–190, New York, NY, USA, 2006. ACM.
11. Y. Wang, L. Aroyo, N. Stash, and L. Rutledge. Interactive User Modeling for Personalized Access to Museum Collections: The Rijksmuseum Case Study. In *Proceedings of User Modeling 2007*, pages 385–389, Corfu, Greece, June 2007.

[9] `http://sourceforge.net/projects/duine`