In-Depth Analysis of Similarity Knowledge and Metric Contributions to Recommender Performance *

Derry O'Sullivan and **Barry Smyth**
Smart Media Institute
University College Dublin
{dermot.osullivan,barry.smyth}@ucd.ie

David C. Wilson
Dept. of Software and Information Systems
University Of North Carolina at Charlotte
davils@uncc.edu

Abstract

Initial successes in the area of recommender systems have led to considerable early optimism. However as a research community, we are still in the early days of our understanding of recommender systems. Evaluation metrics continue to be refined but we still need to account for the relative contributions of the various knowledge elements that play a part in the recommendation process. In this paper, we make a fine-grained analysis of a successful approach in the area of case-based recommendation, providing an ablation study of similarity knowledge and similarity metric contributions to improved system performance. We gauge the strengths and weaknesses of knowledge components and discuss future work as well as implications for research in the area.

Introduction

The late 1990's saw a growing interest in the use of so-called recommender systems as a way of helping users to deal with ever-increasing information overload (Resnick & Varian 1997). Since the earliest recommender systems there have been an abundance of algorithmic developments that have led to a variety of different basic recommendation techniques and strategies. For example, content-based recommendation techniques rely on the availability of meta-data that captures the essence of the items available for recommendation—a movie recommender might make use of movie descriptions that include genre, actor and director information—taking advantage of similarity assessment techniques to match a target user's profile to a set of recommendable items (Rosenstein & Lochbaum 2000; Basu, Hirsh, & Cohen 1998; Soboroff & Nicholas 1999; Smyth & Cotter 2001). Collaborative techniques, such as automated collaborative filtering (Konstan *et al.* 1997; Smyth & Cotter 2001; Terveen *et al.* 1997), provide an alternative strategy in which the meta-data descriptions are sacrificed in favour of ratings-based user profiles. Collaborative filtering (CF) identifies suitable items for recommendation not because their description matches them with a target user but rather because these items have been liked by users who are similar to the target user; a collaborative filtering movie recommender 'knows' nothing about a movie's genre or actors or its director, but it knows that other users have liked this movie and that these users are similar to the target user in the sense that they and the target users have liked and disliked many of the same movies in the past. Thus in general, content-based methods rely on item-item (Sarwar *et al.* 2001) and item-user (Sarwar *et al.* 2000) similarities whereas collaborative filtering methods rely on user-user similarities (Konstan *et al.* 1997).

Although recommender systems have provided a rich vein of research, there are still significant gaps in our knowledge when it comes to a detailed understanding of the computational strengths and weaknesses of specific techniques. The evaluation work that has been conducted to date has largely taken the form of a coarse-grained accuracy or precision-recall analysis (e.g., (Konstan *et al.* 1997; Smyth & Cotter 2001)) without providing a fine-grained assessment of the individual elements (similarity knowledge sources, matching functions, ranking metrics etc.) that make up a particular recommendation strategy. As a result it is often unclear as to which of these elements contribute more or less to the performance of a recommender in a given application scenario.

In this paper we present a detailed and fine-grained performance analysis of similarity knowledge elements that underlie our work in case-based recommendation (O'Sullivan, Wilson, & Smyth 2002; 2003). We focus here on our recent research into the use of data-mining techniques to drive a novel case-based recommendation technique, which is summarized in the next section. We describe the results of a comprehensive ablation study that seeks to identify the key sources of competence and performance that exist within this system by manipulating the similarity knowledge and ranking functions used by our system in order to fully characterize their performance implications. We hope that this paper will serve not only as a further source of evaluation detail on our own work, but also act as a call for other researchers to provide their own ablation studies so that

*The support of the Informatics Research Initiative of Enterprise Ireland is gratefully acknowledged.

our community can better understand the implications of past, present and future recommender systems developments.

Case Based Recommendation

Our recent work in case-based recommendation has applied data mining techniques to derive similarity knowledge in order to ameliorate similarity-coverage problems that arise for systems employing ratings-based user profiles as cases. Issues of similarity coverage arise from the relative sparsity of ratings overlap between average user profiles. Our case-based approach addresses the sparsity problem by first applying data-mining techniques to a set of ratings-based user profiles in order to derive similarity knowledge in the form of rules that relate items. As we will see in the following subsections, these item-item rules and their associated probabilities are used to increase the density of the user-item ratings matrix by leveraging similarities between profile cases to reduce ratings sparsity. Due to space constraints, and to avoid repetition, these sections provide only a technical summary of our case-based technique and the interested reader is referred to (O'Sullivan, Wilson, & Smyth 2002; 2003) for additional detail. Examples in the following discussion are taken from the PTVPlus television programme recommendation domain, described in the next section.

Association Rule Mining

The Apriori algorithm (Agrawal *et al.* 1996) is a well-known data-mining technique that can be used to efficiently discover similarity knowledge from PTVPlus profile cases by finding frequently occurring associations between rated profile items (television programmes), and by assigning confidence scores to the associations. These association rules indicate which items can be considered to be similar, and their associated confidences can be used as a proxy for their level of similarity. In turn, these *direct* rules can be chained together to produce additional *indirect* associations and similarities in order to further elaborate the item-item similarity matrix. When mining association rules, confidence and support values are used to constrain exponentially large candidate rule sets by setting appropriate thresholds. The Apriori algorithm is designed to efficiently process a database of transactions to discover well-supported association rules by finding the set of most frequently co-occurring items (O'Sullivan, Wilson, & Smyth 2002). We should emphasize that data mining using the Apriori algorithm is one of many possible approaches to generating additional similarity knowledge; we have simply chosen data mining as a reasonable initial technique to demonstrate the feasibility of our new recommendation strategy.

Direct & Item-Item Similarities

By treating PTVPlus user profiles as transactions and the rated programmes therein as itemsets, the Apriori algorithm can be used to derive a set of programme-programme association rules with confidence values serving as similarity scores. For example, in PTVPlus we might find the rule *Friends* $\Rightarrow$ *ER* with a confidence of 37%, allowing us to conclude a 37% similarity between *Friends* and *ER* to fill the appropriate slot in our similarity matrix.

These direct associations can be chained together to further improve similarity coverage. For example, discovering rules $A \Rightarrow B$ and $B \Rightarrow C$ may indicate that A and C are also related and the strength of their relationship can be estimated by combining their individual confidence values (see (O'Sullivan, Wilson, & Smyth 2002) for further details). Experiments in this paper use a maximal combination model to calculate indirect rule confidences.

Recommendation Strategy

The recommendation strategy consists of two basic steps:

1. The target profile, t is compared to each profile case, $c \,\epsilon\, C$, to select the k most similar cases.
2. The items contained within these selected cases (but absent in the target profile) are ranked according to the relevance to the target, and the r most relevant items are returned as recommendations.

Profile Matching: The profile similarity metric (Equation 1) is computed as the weighted-sum of the similarities between items in the target and source profile cases. If there is a direct correspondence between an item in the source, c_i, and the target, t_j, then maximal similarity is assumed (Equation 2). However, direct correspondences are rare and so the similarity value of the source profile item is computed as the mean similarity between this item and the n most similar items in the target profile case $(t_1, ..., t_n)$ (Equation 3).

$$PSim(t, c, n) = \sum_{c_i \epsilon c} w_i \cdot ISim(t, c_i, n) \tag{1}$$

$$ISim(t, c_i, n) = 1 \; if \; \exists \, t_j = c_i \tag{2}$$

$$= \frac{\sum_{j=1..n} sim(t_j, c_i)}{n} \tag{3}$$

Recommendation Ranking: Once the k most similar profile cases $(\hat{C})$ to the target have been identified, their items are combined and ranked for recommendation using three criteria. We prioritize items that (1) have a high similarity to the target profile case, (2) occur in many of the retrieved profile cases, and (3) are recommended by profiles most similar to the target. Accordingly we compute the *relevance* of an item, c_i, from a retrieved profile case, c, with respect to the target profile, t, as shown in Equation 4; where $C' \subseteq \hat{C}$ is the set of retrieved profile cases that contain c_i.

$$Rel(t, c_i, \hat{C}) = ISim(t, c_i, k) \cdot \frac{|C'|}{|\hat{C}|} \cdot \sum_{c \epsilon C'} PSim(t, c, k) \tag{4}$$

Finally, the top-N ranked items are returned for recommendation; for these experiments, we have selected an N value of 10 recommendations.

A Fine-Grained Ablation Study

Our previous work has shown that this approach is quite successful in addressing the sparsity problem to produce (1) higher-quality recommendations and (2) better orderings of recommendation results, especially when compared to traditional collaborative filtering approaches (O'Sullivan, Wilson, & Smyth 2002; 2003). Intuitively, we expect that the main contributing success factors are to be found in the derived similarity knowledge that provides additional similarity coverage and in the ranking metric that applies the derived knowledge for ordering recommendations. Each of these components, however, can be analysed at a deeper level, and here we are interested in characterizing the relative contributions of their constituent elements. We do so by performing ablation studies on these components. This analysis may provide a clearer view of the essential strengths of the approach, as well as insights that would be useful in developing more effective recommendation systems.

The ranking metric provides a natural breakdown for analysis in its component factors and the possibilities for their interaction. Here we analyse the contributions of the individual factors and their possible combinations toward good recommendation ranking. It is more difficult to characterize relative contributions of the derived similarity knowledge components. At an atomic level, the additional similarity knowledge consists of item associations, and we adopt the view that selectively testing the contribution of clusters of such associations, based on a measure of their reliability, can provide insight into the power of the applied whole. Thus we propose comprehensive real-world tests that individually focus on:

- The importance of the quality of the mined similarity knowledge relative to recommendation accuracy;
- The overall importance of the similarity knowledge combined with different ranking factors relative to recommendation accuracy.

Datasets

We conduct our experiments using a dataset from the television domain; PTVPlus (www.ptvplus.com) is an established online recommender system deployed in the television listings domain (Smyth & Cotter 2001). Operated commercially by ChangingWorlds (www.changingworlds.com), PTVPlus uses its recommendation engine to generate a set of TV programme recommendations for a target user, based on their profiled interests, and it presents these recommendations in the form of a personalized programme guide. We use the standard PTVPlus dataset consisting of 622 user profiles, extracting a list of positively rated programmes from each profile for use in our system. We have ignored the negative ratings and also the rating values themselves, leaving these factors for future work.

Algorithms

We use a number of different algorithms in testing the aforementioned recommender strategies - both direct and indirect similarity knowledge is used as well as varying the criterion used in recommendation ranking. A recommendation technique without similarity knowledge is used for baseline comparison.

1. *NOSIM* - System run with only collaborative filtering style similarity knowledge (diagonal matrix of item-item relationships);
2. *DR* - our case-based approach using direct similarity knowledge with all recommendation ranking criteria;
3. *INDR* - our case-based approach using indirect similarity knowledge with all recommendation ranking criteria.

We also run variants of *DR* and *INDR* which only use some of the recommendation ranking criterion (example *C1&2-INDR* is *INDR* but only uses criteria (1) & (2) in recommendation ranking). This will allow us to see the effect of the recommendation ranking criteria in overall recommendation accuracy as well as ranking accuracy.

Method & Metrics

The dataset is split into test and training subsets using a 30:70 split. Using the Apriori (Agrawal *et al.* 1996) technique, we generate rules from the training data and then analyse these rules to see how well the similarity knowledge fits the test dataset (O'Sullivan, Wilson, & Smyth 2002); we do this by counting the percentage of profiles that a given rule 'fits' in the sense that the antecedent and consequent (both rule items) are both in the profile. To see the effect of rule accuracy on the overall quality of system recommendations, we sort rules by how well they fit the test dataset, placing them into bins of rules less than 10% accurate, less than 20% accurate and so on, up to less than 70% accurate (highest rule accuracy seen on test dataset).

We are also interested in testing the quality and ranking of our techniques. We take the full association ruleset (direct rules) created from the training dataset and extended to a indirect ruleset. Using the different algorithms described earlier, we then tested both recommendation quality and ranking quality of the system.

In evaluating recommendation accuracy, our primary accuracy metric measures the percentage of test profile items that are present in a user's recommended set; this is equivalent to the standard recall metric used in information retrieval. So, for example, if all of the items in a user's test profile are contained within their recommended set a maximum recall of 100% is achieved.

Recall: The proportion of items in the user's test profile that are recommended, averaged over all users.

In general recall is a strong measure of recommendation accuracy and it should be noted that in our evaluation it serves as a *lower-bound* on real recommendation accuracy. This is because the only way that we can judge a recommendation to be relevant is if it exists in the user's test profile, which of course represents only a limited subset of those recommendations that are truly relevant to the user. With this in mind we also introduce a weaker notion of recommendation accuracy, which we call *hit rate*. The basic idea is that a given set of recommendations has at least some measurable value of usefulness to a user when it contains at least one recommendation from the user's test profile. A maximum hit rate of 100% indicates that a given algorithm always makes at least one relevant recommendation (present within the user's test profile) per recommendation session.

Hit Rate: The proportion of users for which at least one item from the user's test profile is recommended.

To test our ranking functions, we look for correlations between the rank of an item and its recommendation success over all profiles. Programmes are ranked by recommendation list position and sorted into ranking bins (0 - 1.0 in 0.1 increments). For example, if we have a list of 20 recommendations then these are distributed across the 10 bins with 2 recommendations in each. We repeat this binning for each recommendation list produced from a profile and calculate the percentage of correct recommendations in each bin. We are looking for a strong positive correlation between the success percentage in a bin and the ranking value of that bin; higher ranking values should lead to higher percentages of correct recommendations.

Results

We conduct two experimental studies; the first analyses the relative contributions of clusters of similarity knowledge, and the second examines the relative contributions of factors in recommendation ranking.

Similarity Knowledge

The association rules that comprise the similarity knowledge are categorised by their data-set accuracy and clustered into bins at decimal percentage levels. Experimental runs are then made with progressively increasing levels of similarity knowledge by augmenting the current set of associations with those from the next higher set. At each level, system accuracy is measured in order to determine the relative uplift in accuracy provided by the additional knowledge. Figure 1 shows the results for Recall, and Figure 2 shows the results for Hit Rate.

At first glance, the results show a natural trend of increasing accuracy as similarity knowledge grows, as expected. However, closer inspection reveals a few surprises. For both Recall and Hit Rate measures, the uplift provided by adding additional associations

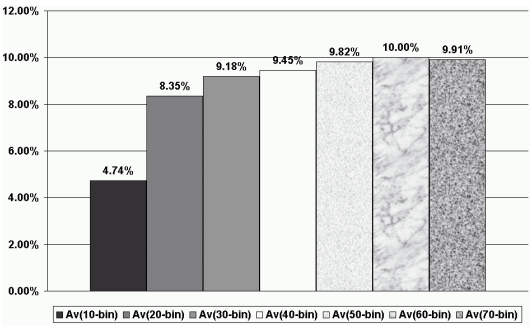

Figure 1: Rule Binning Recall

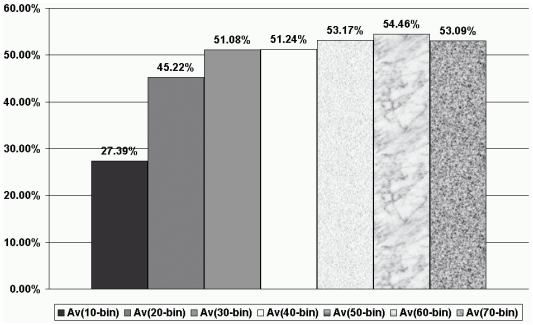

Figure 2: Rule Binning Hit Rate

levels off very rapidly, so that the relative contribution of an additional associations is fairly small above the 30%-bin level. This indicates that the addition of higher-probability, presumably better quality, associations offers only slight improvement to the set of lower-probability associations. It may be that the set of lower-probability associations as an ensemble can provide a significant proportion of the potential additional similarity coverage, a sort of boosting effect. It may also be that some associations interfere with one another in computing similarities, which limits the impact of additional knowledge. An interference effect is likely, as the addition of the highest-probability association set actually produces a slight decrease in performance.

From the standpoint of implementing recommender systems, these results indicate that it may be possible to achieve a significant boost in performance with (1) a smaller proportion of the knowledge that can be discovered and (2) comparatively weaker associations. The former has implications for efficiency, and the latter has implications for applicability of the technique, even for datasets where only weak associations can be derived.

Recommendation Ranking

Rankings and accuracies were computed for each combination of the three factors in the item ranking criteria:

- C_1 - Item similarity to target profile

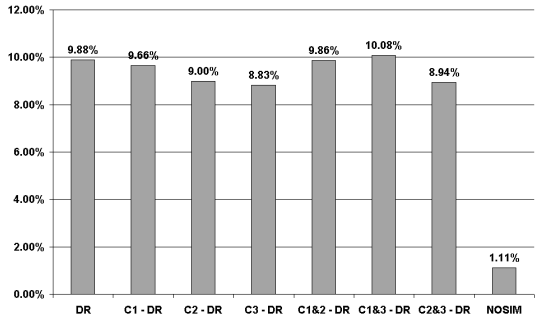

Figure 3: Direct Criteria Recall

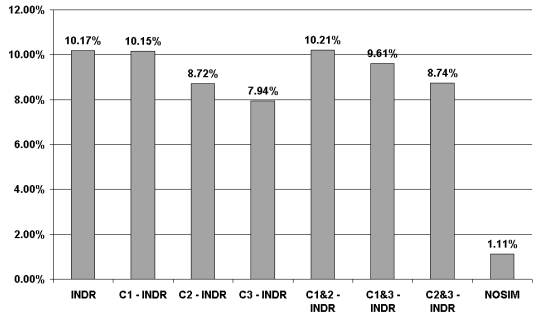

Figure 4: Indirect Criteria Recall

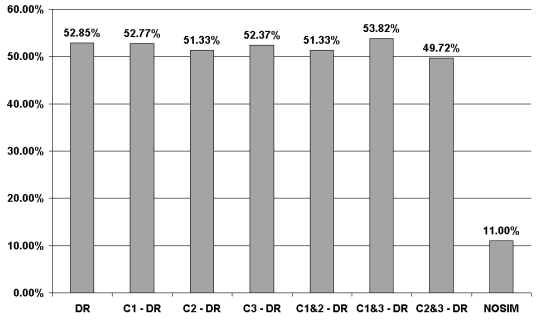

Figure 5: Direct Criteria Hit Rate

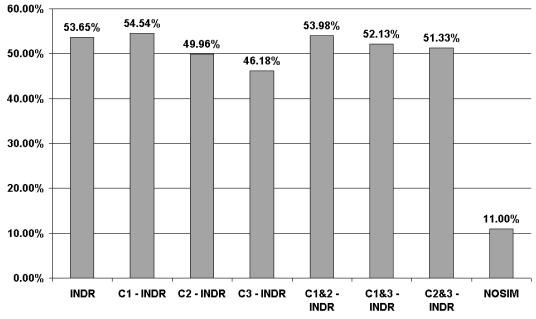

Figure 6: Indirect Criteria Hit Rate

- C_2 - Item frequency across retrieved profiles
- C_3 - Similarity of item profile to target profile

These combinations were computed both for direct and indirect modes of employing derived rule associations. Accuracies were measured in terms of Recall, Hit Rate, and Rank Binning Correlation. In terms of system accuracy, Figures 3 and 4 show the results for Recall, and Figures 5 and 6 show the results for Hit Rate. Ranking makes a difference in system accuracy, since the similarity metric provides the top k similar profiles from which a variable number of candidate items may be derived for recommendation. After the ranking is applied, only the top N (10 here) items are taken as actual recommendations, which can impact accuracy.

Employing similarity knowledge is clearly beneficial in all combinations, compared to the NOSIM baseline. Across all combinations, direct and indirect modes perform similarly, with minor variations. This is consistent with earlier comparisons between direct and indirect modes, but it is verified here across the finer-grained analysis, showing that each of the ranking criteria provides similar benefit across direct and indirect modes. Individually, C_1 outperforms the others, with C_2 outperforming C_3 in most conditions. In combination, it can be seen that C_1 is more consistent. Combining C_1 with the others tends to result in an improvement, while combining another with C_1 tends to result in a degradation of performance. The differences between the best performances are very small indeed, but it is worth noting that, while the best performance combination is not consistent, the standard mode of system operation (combining all 3 criteria) is consistently within 1% of the best, and the best performing combination consistently involves C_1.

In terms of the rank binning correlations, shown in Figures 7 and 8, we again find that indirect and direct modes show similar overall results. Most of the combinations provide very good and highly comparable rankings. The notable exceptions are C_2 and the standard system combination of all 3. Again C_1 appears in the best combinations. Overall, we find that C_1 is the most consistent indicator of good ranking performance, and that while the standard system combination of all 3 criteria is consistent in the overall ranking for providing accuracy, it is not as consistent in ordering the final recommendation set. In terms of implementing recommender systems, it may be worth focusing on C_1 for recommendation ranking.

Overall, these results indicate that the success of our approach may be based more on underlying finer-grained critical success factors than have previously been thought. By refining the similarity knowledge to eliminate possible interference in the associations, we might hope to improve performance further, or at least to improve efficiency by streamlining the similarity knowledge set for a given performance level. By focusing on the most important ranking criteria, it may also be possible to improve efficiency while maintaining the same level of performance.

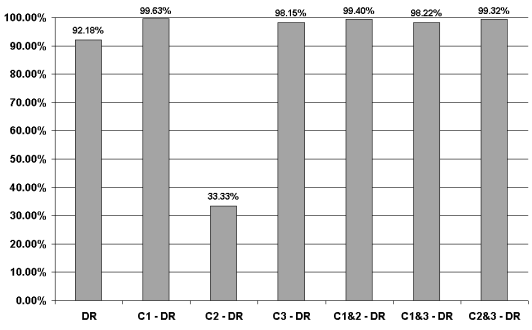

Figure 7: Direct Rank Binning Correlation

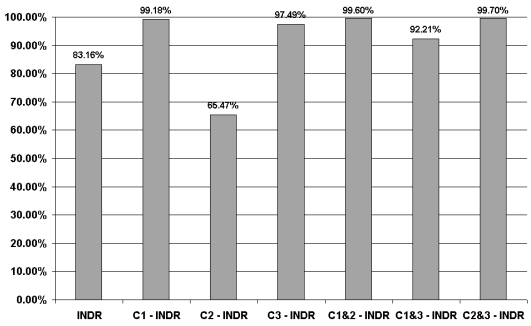

Figure 8: Indirect Rank Binning Correlation

Conclusions and Future Work

In this paper, we have presented a new and significantly deeper analysis and evaluation of a recently developed and proven recommender approach, in order to determine critical underlying contributions to system performance as a whole. Our aim was to highlight the need for further evaluation of existing techniques in order to expose contributions and developments that have so far remained hidden in the coarser-grained evaluations. To this end we have described a fine-grained ablation study of our own case-based recommendation approach, which has revealed a number of interesting and surprising results. These results have implications for the recommender research community in general, such as the value of weak association rules and the dominance of certain ranking criteria. The PTVPlus data set has proven a good indicator of performance across other domains, but we expect to validate results in additional data sets. We plan to refine the analysis to further examine the functional characteristics of the derived similarity knowledge, with an eye toward a coverage model of similarity as with the case-base knowledge container. We also plan to test the refinements suggested here for incorporation as part of the standard system model. In closing we hope that fellow researchers will continue to cast a critical eye over their own research and pay close attention to potential fundamental underlying factors that drive their own systems towards improved recommendation performance.

References

Agrawal, R.; Mannila, H.; Srikant, R.; Toivonen, H.; and Verkamo, A. I. 1996. Fast Discovery of Association Rules. In Fayyad, U. M.; Piatetsky-Shapiro, G.; Smyth, P.; and Uthurusamy, R., eds., *Advances in Knowledge Discovery and Data Mining*. AAAI Press. chapter 12.

Basu, C.; Hirsh, H.; and Cohen, W. W. 1998. Recommendation as Classification: Using Social and Content-Based Information in Recommendation. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence, Madison, Wisconsin, USA, July 1998*, 714–720. AAAI Press.

Konstan, J. A.; Miller, B. N.; Maltz, D.; Herlocker, J. L.; Gordon, L. R.; and Riedl, J. 1997. Grouplens: Applying Collaborative Filtering to Usenet News. *Communications of the ACM* 40(3):77–87.

O'Sullivan, D.; Wilson, D.; and Smyth, B. 2002. Using Collaborative Filtering Data in Case-based Recommendation. In Haller, S. M., and Simmons, G., eds., *Proceedings of the Fifteenth International Florida Artificial Intelligence Research Society Conference*, 121 – 128. AAAI Press.

O'Sullivan, D.; Wilson, D.; and Smyth, B. 2003. Preserving Recommender Accuracy and Diversity in Sparse Datasets. In Russell, I., and Haller, S., eds., *Proceedings of the Sixteenth International Florida Artificial Intelligence Research Society Conference*, 139 – 144. AAAI Press.

Resnick, P., and Varian, H. R. 1997. Recommender Systems. *Communications of the ACM* 40(3):56–58.

Rosenstein, M., and Lochbaum, C. 2000. Recommending from Content: Preliminary Results from an E-Commerce Experiment. In *CHI '00 Extended Abstracts on Human Factors in Computer Systems*, 291–292. ACM Press.

Sarwar, B.; Karypis, G.; Konstan, J.; and Riedl, J. 2000. Analysis of Recommendation Algorithms for E-Commerce. In *Proceedings of the 2nd ACM Conference on Electronic Commerce*, 158–167. ACM Press.

Sarwar, B.; Karypis, G.; Konstan, J.; and Reidl, J. 2001. Item-based Collaborative Filtering Recommendation Algorithms. In *Proceedings of the Tenth International Conference on World Wide Web*, 285–295. ACM Press.

Smyth, B., and Cotter, P. 2001. Personalized Electronic Programme Guides. *Artificial Intelligence Magazine* 22(2):89–98.

Soboroff, I., and Nicholas, C. 1999. Combining Content and Collaboration in Text Filtering. In *Proceedings of the IJCAI-99 Workshop on Machine Learning for Information Filtering*.

Terveen, L.; Hill, W.; Amento, B.; McDonald, D.; and Creter, J. 1997. PHOAKS: A System for Sharing Recommendations. *Communications of the ACM* 40(3):59–62.