Chapter 6

Recommender Systems for the Web

J. Ben Schafer, Joseph A. Konstan, and John T. Riedl

6.1 Introduction

As the Web rapidly evolved into an immense repository of content, human users discovered that they could no longer effectively identify the content of most interest to them. Several approaches developed for improving our ability to find content. Syntactic search engines helped index and rapidly scan millions of pages for keywords, but we quickly learned that the amount of content with matching keywords was still too high. Semantic annotation helps assist automated (or computer-assisted) processing of content to better identify the real contents of pages. A Semantic Web would help people differentiate between articles on “china” plates and articles about “China” the country. Recommender systems tried a different approach—relying on the collected efforts of a community of users to assess the quality and importance of contents. For example, a Web page recommender may identify that a particular page is popular overall, or better yet, popular among people with tastes like yours.

Recommender systems are generally independent of syntactic and semantic approaches. Hence, recommenders are often combined with other technologies. For example, the search engine Google (www.google.com) combines syntactic indexing with a nonpersonalized recommender (the PageRank algorithm) that assigns higher value to pages that are linked to by high-value pages. Similarly, most recommender system interfaces allow users to first use search techniques to narrow down the set of candidates being evaluated.

The term “recommender systems” was first coined at a workshop in 1996, and has been used imprecisely and inconsistently in published work. For this chapter, we limit ourselves to discussing systems that have two key properties: (1) they incorporate the experiences or opinions of a community, and (2) they produce for the user a mapping of scores to items, a set of recommended items, a ranking of items, or a combination of these three. We further focus on personalized recommenders—those that produce different recommendations for different users—and automated recommenders—those that do not require explicit actions to send or receive recommendations. These are the recommender systems that have been most prominent and successful in industrial applications (Schafer et al., 2001).

Such recommenders have not only been successful commercially, but they have been shown to be valuable in experimental research. Early studies of recommenders in Usenet news showed that users valued personal recommendations for articles and that they were nearly twice as likely to read an article with a high recommendation

score compared with one with a low score (Miller et al., 1997). More recent studies have shown that the displayed recommendation score also influences the user's subsequent rating of items that are recommended, and that user satisfaction with recommender systems suffers when recommendations are randomly perturbed to create a less accurate recommender (Cosley et al., 2003).

This chapter provides a survey of research on recommender systems, focusing on recommenders that can be applied to the Web. While reflecting on research that was largely independent of the Semantic Web, it looks at how recommenders and the Semantic Web can be integrated to provide better information sources for users. Finally, it reviews a number of user interface issues related to recommenders on the Web.

6.2 The Beginning of Collaborative Filtering

As content bases grew from mostly “official” content, such as libraries and corporate document sets, to “informal” content such as discussion lists and e-mail archives, users began to experience a new type of overload. Pure content-based techniques such as information retrieval and information filtering were no longer adequate to help users find the documents they wanted. Keyword-based representations could do an adequate job of describing the topic of documents, but could do little to help users understand the nature or quality of those documents. Hence, a scholarly report on economic conditions in Singapore could be confused with a shallow message by a recent visitor who speculates that “the economy is strong—after all, I paid over \$12 for a hotel breakfast!” In the early 1990s there seemed to be two possible solutions to this new challenge: (1) wait for improvements in artificial intelligence that would allow better automated classification of documents, or (2) bring human judgment into the loop. Sadly, the AI challenges involved are still formidable, but fortunately human judgment has proved valuable and relatively easy to incorporate into semiautomated systems.

The *Tapestry* system, developed at Xerox PARC, took the first step in this direction by incorporating user actions and opinions into a message database and search system (Goldberg et al., 1992). *Tapestry* stored the contents of messages, along with metadata about authors, readers, and responders. It also allowed any user to store annotations about messages, such as “useful survey” or “Phil should see this!” *Tapestry* users could form queries that combined basic textual information (e.g., contains the phrase “recommender systems”) with semantic metadata queries (e.g., written by John OR replied to by Joe) and annotation queries (e.g., marked as “excellent” by Chris). This model has become known as *pull-active collaborative filtering*, because it is the responsibility of the user who desires recommendations to actively pull the recommendations out of the database.

Tapestry predates the Web, let alone the Semantic Web, but it carries several important lessons. First, there is a great deal of information to be gained from even simple semantic markup. Being able to select pages based on authors, modification dates, and other commonly available fields makes it easier for users to find the pages they want. Second, usage patterns from other users can be valuable, especially if those users are trusted. Pull-active collaborative filtering requires a community of people who know each other well enough to decide which opinions matter; I need to know that Chris’s “excellent” means something, while Jean’s may not. Finally, *Tapestry* reminds us that after a certain point, it becomes more effective to add deep (in this

case human) semantic annotations, rather than wait for the technology to improve automated syntactic and surface semantic processing.

Soon after the emergence of Tapestry, other researchers began to recognize the potential for exploiting the human “information hubs” that seem to naturally occur within organizations. Maltz and Ehrlich (1995) developed a *push-active collaborative filtering* recommender system that made it easy for a person reading a document to push that document on to others in the organization who should see it. This type of push-recommender role has become popular, with many people today serving as “joke hubs” who receive jokes from all over and forward them to those they believe would appreciate them (though often with far less discriminating thought than was envisioned).

Tacit Corporation’s ActiveNet (www.tacit.com) extends these ideas into a broader knowledge-management system. The system combines traditional information filtering with a push-distribution interface and query system to find experts. The system builds private profiles of each user’s interests based on keyword analysis of the content the user generates during daily work. Depending on the source of these keywords (e.g., public documents vs. private emails) differing privacy levels are assigned to keywords, and users can override these initial settings using the Profile Management Interface

Figure 6.1 Tacit’s KnowledgeMail provides a Profile Management Interface through which users can modify the privacy levels of their profile. Other users can push content to these users based on the public portions of a profile.

(Figure 6.1). The push-distribution interface allows the sender of a message to ask the system to notify appropriate recipients based on interest. In addition, users can seek expertise by asking the system to contact experts on the topic of a particular message. This hybrid of content and manual collaborative filtering can help an organization be more efficient in distributing information to those who need it without overwhelming everyone with everything.

So far, push-active recommenders lack the rich semantic annotation promised in the Semantic Web. There is great potential for improvement of these systems if semantic tagging could be included—but only if it is automated. For example, if automated tagging could identify business-related semantic entities such as competitors, markets, products, etc., messages could almost certainly be routed more automatically, and profiles could be structured more sensibly and compactly.

6.3 Automated Collaborative Filtering

A limitation of active collaborative filtering systems is that they require a community of people who know each other. Pull-active systems require that the user know whose opinions to trust; push-active systems require that the user know to whom particular content may be interesting. Automated collaborative filtering (ACF) systems relieve users of this burden by using a database of historical user opinions to automatically match each individual to others with similar opinions. Another advantage of ACF systems is the ability for users to participate using a pseudonym, and therefore without revealing their true identity.

Intuitively, ACF systems follow a process of gathering ratings from users, computing the correlations between pairs of users to identify a user's "neighbors" in taste space, and combining the ratings of those neighbors to make recommendations. More formally, the early ACF systems—GroupLens (Resnick et al., 1994), Ringo (Shardanand and Maes, 1995), and Video Recommender (Hill et al., 1995)—all used variants of a weighted k -nearest neighbour (NN) prediction algorithm.

Since the construction of these early ACF systems, a variety of algorithmic research has been conducted that attempts to refine or extend the basic NN algorithm. Herlocker et al. (1999) studied the effect on accuracy of the various refinements that could be made to the tradition NN algorithm including various similarity correlations, weighting techniques, methods for selecting neighborhoods, and the use of normalization of ratings to reflect different user rating schemes. Sarwar et al. (2001) extend the NN algorithm to correlate items-to-items, which can be helpful when there are many more users than items. Deshpande and Karypis (2004) have confirmed that such systems are as accurate as user-based techniques while producing recommendations up to two orders of magnitude faster. McLaughlin et al. (2004) have modified the algorithm to include belief distributions, which can be used to better differentiate between highly rated items by weakly correlated users and "average" rated items by strongly correlated users.

Other algorithmic research has focused on significantly different approaches. Breese et al. (1998) studied the generation of Bayesian networks with decision trees at each node of the structure. Wolf et al. (1999) generated recommendations through a similarity graph-based approach known as horting. Sarwar et al. (2000) generated more efficient recommenders by considering the application of dimensionality reduction through singular value decomposition. Jin et al. (2003) build a system that employs two complementary models, one of a user's preferences, and the other of a user's rating

scheme, and use a Bayesian algorithm to combine the predictions produced across varying models.

With the variety of algorithms that can be applied to the generation of recommendations in ACF systems, it is important to consider mechanisms for evaluating different systems. Herlocker et al. (2004) studied the different metrics different practitioners have used to determine if one system is “better” than another. In doing so, they have discovered large differences in techniques and rationale. Evaluation techniques vary based on the types of task being evaluated and the types of data used in generating predictions. Furthermore, metrics vary from those evaluating the pure quality of the predictions to those that attempt to analyze additional attributes such as user satisfaction and the ability to make decisions with confidence. They have concluded that metrics can be collapsed into three equivalence classes where metrics within each class correlate strongly with each other, but weakly with those from other classes.

Algorithmic research for pure ACF has slowed in recent years as the change in accuracy from these new algorithms has decreased significantly. Instead, research has focused on algorithms for systems with very different objectives, such as algorithms for hybrid systems, or those designed to support specific interfaces or unusual data sets. This type of research is presented in more detail in future sections.

Interfaces for recommender systems attempt to visually present recommendations in a manner useful to users.

The GroupLens recommender system (Resnick et al., 1994; Konstan et al., 1997) used a very explicit interface where ratings were entered manually by keystroke or button, and ratings were displayed numerically or graphically (Figure 6.2). Improvements to this type of interface have taken two forms: (1) less visible interfaces such as the implicit ratings discussed below and the use of subtle suggestions (e.g., product placement) in place of explicit predictions and identified recommendations, and (2) more information-intensive interfaces, which have generally been used to display additional data to help users make decisions, such as Ringo’s community interaction interfaces and the explanations interfaces presented below.

Commercial users of recommender systems may classify interfaces according to the types of recommendations made. *Predictions* are forecasts of how much a user will like an item, and can be used in response to queries, as annotations to be used while browsing, or in combination with other techniques. *Suggestions* are lists of items, or individual items, presented to a user as recommended, often these are associated with the top levels of a site or a personalized “my site” page. *Item-associated* recommendations are displayed in conjunction with specific items, usually based on evidence that those who liked or purchased the item being viewed also liked the recommended ones. Finally, *organic* interfaces often remove the evidence of the recommendation, making the site simply appear natural, yet personalized. For example, a custom Web-based newspaper may rearrange stories to place ones with high predicted interest at the front (Bharat et al., 1998).

The basic approach of automated collaborative filtering has been developed quite apart from the semantic tagging and analysis that is emerging in the Web. Indeed, one of the goals of early collaborative filtering was to be “content-neutral,” and indeed the early ACF software was completely ignorant of all content attributes. Later work has shown that this approach was too pure; humans care about content, and both their ratings and their usage of recommender systems can be content-specific. Accordingly, later systems, including commercial systems, support the definition of content attributes that can be used both to partition the space of items (to define subspaces with better correlations among users) and to support interfaces that allow

Figure 6.2 A modified Xrn news reader. The GroupLens project added article predictions (lines of ### on the top right) and article rating buttons (bottom).

users to overlay content and quality questions. A simple example of this approach is found in MovieLens, which allows users to receive recommendations for movies by genre and date.

Looking forward, however, there are areas of synergy between automated collaborative filtering and the Semantic Web. First, semantically tagged content can be better classified, and ratings of it can be better analyzed to build models of user interest. Second, user opinions *are* semantic evaluations of content; they can be processed to supplement other techniques for semantic analysis. For example, if a semantic tagger cannot tell whether a particular Web page refers to the old New York Giants baseball team, or the New York Giants football team, perhaps because the content is sufficiently well in the overlap, human ratings could well disambiguate the reference. Finally, automated collaborative filtering is still useful in a Semantic Web, and may add more value to users when a topic is already well-established.

Interesting examples of such overlap occur in the Platform for Internet Content Selection (PICS) and the Platform for Privacy Preferences (P3P). PICS was designed to provide a standard interface through which Web browsers could access rating bureaus,

and through which rating bureaus could provide page ratings for their subscribers (Resnick and Miller, 1996). The idea was to unify the wide range of content rating and filtering services that were emerging with a standard interface, so that those who wanted to filter out pornography or immorality could do so, and so that others who might want to rate pages on other criteria (including automated collaborative filtering) could set up such rating bureaus. Though the rapid explosion and pace of the Web dissuaded many would-be ratings services, today we can think of PICS annotations as yet another source of semantic annotations for Web pages and links. Indeed, a Semantic Web browser today may choose to consult the filtering services that identify content inappropriate for minors so that they can alter the display of a page (by omitting such links or obscuring inappropriate images, for example).

P3P grew out of PICS and was designed to provide a standard interface through which Web sites can communicate about their privacy policies (Cranor, 2002). “Traditional” policy statements give consumers access to information about a site’s privacy policies. These policies are written in a human-readable language that can be confusing due to human error in editing or purposefully written to contain room for interpretation. As such, users must decipher the language of the policy to determine if the practices are acceptable to them and whether they choose to do business with a site. P3P provides a way for Web sites to present their privacy policies in a standardized and machine-readable format by answering a series of multiple-choice questions that clearly place the site within one of several “buckets.” By building P3P tools into browsers and other Web-based applications, users can configure a set of user preferences for the type of data they are willing to share and the circumstances under which these data may be shared. Web tools can compare the published privacy policies of visited sites with a users preference profile and offer advice or alerts when users encounter sites whose privacy practices may not meet their comfort level.

6.4 Enhancing Collaborative Filtering with Semantics

Content, and in particular semantic content, can be used to address two of the major problems facing collaborative filtering recommender systems: the startup problem and the problem of changing moods. This section explores three different approaches to using content within ACF recommender systems.

6.4.1 New Users and New Items

One opportunity to improve collaborative filtering systems is during the startup process for new users or new items. Collaborative filtering systems cannot recommend items that have never been rated by any user because they have no data on which to base a recommendation. Similarly, collaborative filtering systems offer little value to new users who have not yet rated items. Designing systems that can effectively obtain data from new users or about new items should improve the overall accuracy and helpfulness of recommenders.

Several projects from the GroupLens Research group have considered methods for more efficiently gaining data from new users. Rashid et al. (2002) studied six techniques that CF systems can use to select which items to ask a user about to gain information about a new user. These include entropy, selecting the items that will tell the system the most about the user but may be less likely to have actually been

viewed by the user; popularity, which are the items most likely to have been viewed by a user but may produce less value as a predictive tool; a combination of these two; and item-to-item prediction, a technique that uses the already-given ratings to calculate the items most likely to have been seen by the user based on item-item correlations. Results suggest that the technique to choose depends on the domain as well as the type of user experience the system wants to provide. In a study in the MovieLens movie recommender system, item-to-item prediction identified items that the user could rate, but that were of low marginal information value. Entropy yielded almost entirely obscure films that the user could not rate. The best performance came from a mixture of entropy and popularity.

McNee et al. (2003a) considered interfaces that allow the user to help select the selection of items that are used to develop the initial user model. They conclude that such interfaces have little effect on the accuracy of the system, but do affect user perception of the time and effort spent during the initial rating period and also can affect user loyalty to the system.

Another approach is to use the Semantic Web to help these users find items of interest immediately, according to their past interest profile. For instance, a user navigating to a new site might be recommended items that are labeled the same as items she has liked on other sites she has visited. Over time, the recommendations could be tuned based on the user's specific interactions with the new site in addition to the Semantic Web data. Incorporating both types of recommendations in the same interface would enable users to smoothly switch from startup situations to long-term personalization relationships with Web sites.

Semantic Web information can be used in other ways to improve the performance of collaborative filtering on Web sites. One idea is to use the Semantic Web categories to provide a partitioning of the information on a Web site into categories that should be separately recommended. Experience with collaborative filtering has shown that recommendations made based on data from closely related items are more accurate than recommendations made across broad categories. For instance, if a Web site has medical data on both strength training and cardiovascular fitness, recommendations of content may be more valuable if they are within those categories rather than across categories.

Alternatively, collaborative filtering can be used to identify potential areas for improvement within the Semantic Web labeling for a Web site. For instance, the collaborative filtering engine might point out areas of the site that are rated very similarly though they are far apart in the Semantic Web hierarchy. These differences may point out areas in which the Semantic Web labeling differs from the way users look at data on the site—and may provide an opportunity to improve the usability of the site.

Another possible type of convergence between collaborative filtering and the Semantic Web is the use of collaborative filtering to choose among several possible labelings for a Web site. As the Semantic Web spreads, there are likely to be competing alternatives for labeling popular sites. A new visitor to one of those sites may use collaborative filtering to choose the most useful of the labelings for her.

6.4.2 Integrated Content/Collaborative Filtering Solutions

Combining the Semantic Web and collaborative filtering to solve the startup problem is related to a general approach to collaborative filtering problems in which classical content filtering is combined with collaborative filtering to produce better

recommendations. Pure collaborative filtering is impractical in many domains because of the rate of new items (that arrive unrated) and because of the high sparsity (and high desired sparsity) of the domain. Domains where items are created frequently but have a short useful lifespan are particularly problematic, limiting the effectiveness of recommender systems for news, live discussions, and live or scheduled entertainment. At the same time, there is great interest in recommendations for such content—particularly in the area of news and television programs. Several researchers are examining approaches to better integrate content filtering with collaborative filtering to take better advantage of the strengths of each. Burke (2002) describes several of these approaches including several discussed below, in his survey of hybrid recommender systems.

The GroupLens Research group completed two different studies of automated filtering agents; dubbed Filterbots. Sarwar et al. (1998) considered basic nonpersonalized, text-analysis agents (the length of the articles, the percentage of new content, and the quality of spelling, etc.) in the domain of Usenet news articles. Results showed that different newsgroups and users benefited from different agents, but that overall the agents could be used to improve the recommender system for most users. Good et al. (1999) extended this work by adding a variety of personalized agents for the domain of movies. The agents were trained using rule-induction systems, TFIDF information filtering systems, and basic linear regression on data including genre, cast lists, and descriptive keywords. Results showed that for a small user community (50 users), the best agents outperformed collaborative filtering using only users, but the combination of users and agents in a collaborative filtering system outperformed all other combinations. Later results suggest that as communities get larger, they reach a cut-off point at which agents stop adding much value.

Claypool et al.'s (1999) online newspaper project, P-Tango, creates a personalized newspaper for each user by combining collaborative filtering with user-selected categories and keywords. P-Tango is able to provide better recommendations for users by combining the two techniques than with either technique separately. Figure 6.3 shows the P-Tango front page for an established user.

Fab takes a very different approach to the same goal of combining content and collaborative filtering (Balabonovic and Shoham, 1997). In Fab, user profiles are created using classical content filtering techniques. Distances between neighbors are computed directly on the content-based profiles, rather than on user ratings vectors. These distances are then used by a collaborative filtering algorithm to select neighbors and make predictions in more or less the usual way.

The TechLens Project (Torres et al., 2004) recommends research papers to users by both mapping a Web of citations between papers into the collaborative filtering user-item ratings matrix and analyzing the content of a paper through a TF-IDF approach. Research has considered five differing methods for combining the results of these recommendations. Results of both online and offline studies indicate that different combination algorithms are appropriate in different contexts (is the user looking for papers that are a reasonable introduction to a topic, are the authoritative paper on a topic, present novel innovations on a topic, etc.) but that the combination of data produces results that can aid digital libraries in recommending helpful and meaningful research papers.

Quickstep and Foxtrot (Middleton et al., 2004) also recommend research papers, but do so based on the application of ontological user profiling. Quickstep maintains a personal interest profile for each user based on the topics browsed by users and explicit relevance feedback. More actively, it monitors the URLs visited by users and performs a nightly “offline” ontological classification of these using a computer science research

Figure 6.3 The Tango personalized newspaper combines collaborative and content-based filtering. It provides an appealing rating scale on the right.

paper topic ontology. From this information, Quickstep produces a set of offline recommendations available via the Web browser. Foxtrot added to the functionality of Quickstep by introducing the ability for users to access their profile, the inclusion of a search interface, and e-mail notification of recommendations.

Hoffman (2004) proposes an alternative approach to these techniques by considering the construction of Latent Semantic models for application in collaborative filtering settings. Among other techniques, statistical modeling algorithms in this class of recommenders can be modified to handle the appropriate mixture of explicit user ratings with latent class variables to discover user communities for the generation of recommendations.

All of these techniques can also be applied to the Semantic Web. Filterbots have been proposed as automatic semantic labeling agents for Web sites. The information in the Semantic Web on a site can be used to provide content hierarchies that can be used by systems like P-Tango and TechLens that combine collaborative and content filtering. The Fab approach suggests that neighborhoods can be selected based on other types of data than ratings, such as Semantic Web labels a user tends to prefer. Quickstep and Foxtrot add to the user-defined labels from the Semantic Web to introduce machine-defined ontologies. The Semantic Web may prove a platform for bringing together content and collaborative techniques, improving value to users.

6.4.3 Situational and Task-Focused Recommenders

Collaborative filtering recommenders usually work to find the best recommendations according to a user's complete history of interactions with a Web site. This is, after all,

one of their benefits; they learn what you want based on what you've done in the past. In many cases, though, a recommendation that is based on a less complete history, but that is more appropriate to a particular task, is preferred. For instance, a visitor to a medical Web site might be a long-standing visitor to the site who usually reads expert information about a chronic disease her daughter has been suffering from for years. However, on this particular visit she might be looking for information about a disease her mother has just been diagnosed with. In this case, she does not want the site to steer her toward the chronic disease she usually reads about, and she also does not want to see the expert information she usually reads. Instead, she wants to focus on a new disease, and wants to be treated as a novice. A situational recommender might give her the tools she needs to specify her short-term interest in contrast to her long-term profile.

MetaLens is an example of one approach to a situational recommender (Schafer et al., 2002, Schafer et al., 2004). MetaLens is an extension of the MovieLens framework in which users can specify attributes of the show they want to go to, including its cost, distance to the theatre, show times, appropriateness for children, critic recommendations, and MovieLens prediction. Users have control over which attributes they would like the recommended movies to match, and to which degree matches should be rewarded (or nonmatches penalized). This allows users to indicate that while they are interested in seeing a movie they will enjoy, it may be more important that the movie be appropriate for a 10-year-old and be completed by 9:00 P.M. Figure 6.4 shows a sample MetaLens screen shot of an experiment we did in which users were given specific instructions about a particular situation in which they might need recommendations.

Another situational recommender we have studied in the context of MovieLens enables users to create a set of profiles based on groups of movies (Herlocker and Konstan, 2001). Each profile is a group of movies that have something in common to the user. For instance, one such profile might be "Intelligent Spy Flicks." When the user asks for recommendations, the situational recommender looks for movies that

MetaLens: Preference Screen - Metascope

File Edit View Help Preferences Help

MetaLens: Preference Screen

Movie Features	Preferences	Min. Importance	Max. Importance	Min. Strength	Max. Strength	Display Info
Genre	☐ Action/Adventure ☐ Comedy ☐ Horror ☐ Mystery ☐ Romance ☐ Sci-Fi/Fantasy ☐ Thriller ☐ Drama	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐
Each Feature	☐ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5	☐				

are closely related to those in the group, and that are appropriate according to his long-term interest profile.

One example of a situational recommender outside the context of movies is early work on TechLens. McNee et al. (2002) considered ways to recommend additional citations for a target research paper based on the current citation list for the paper. Offline research considered the formation of such recommendations, while online research studied the user perceived effectiveness of such recommendations.

Situational recommenders like these would benefit from the Semantic Web. The semantic hierarchy on a Web site could serve as information that would be used by a situational recommender to create recommendations appropriate to a particular user at a particular time. For instance, the labels on content at the medical Web site might be used to select content appropriate to the user's newly discovered interest. Collaborative filtering could select among the novice information on the new disease that the user would be most likely to find interesting.

6.5 Explanation and Inference

Recommender systems have two principal interfaces—the recommendation interface and the rating interface. This section discusses one of the more important examples of research to improve each interface by better matching it to user tasks and needs.

6.5.1 Explaining Recommendations

Recommender systems do not exist for their own sake; rather they exist to help users engaged in a task that would otherwise have too many alternatives to adequately consider. In some low-risk domains, users may simply be grateful for the suggestion, and may follow it; in other domains, however, they need to be convinced. In particular, they need to understand enough about a recommendation to decide whether and when to trust it.

One simple mechanism for helping users determine if they can trust a recommendation is through the addition of a confidence metric. While mathematical indicators of confidence exist and could be displayed to users, McNee et al. (2003b) chose to study the effectiveness of displaying simple graphics alongside recommendations for which the system was less confident. Figure 6.5 illustrates how such graphics were used. The single die icon near the movie title *The Horse's Mouth* indicates that it is risky, while the double dice icon alongside *Akahige* indicates that

Recent DVDs	
1. Beautiful Mind, A (2001)	★★★★★
2. Red Beard (Akahige) (1965)	★★★★★
3. From Hell (2001)	★★★★★
4. Traffic (2000)	★★★★★
5. Horse's Mouth, The (1958)	★★★★★

Figure 6.5 Dice icons indicate recommendations that MovieLens considers risky due to limited ratings contained in the system. The single die icon indicates that *The Horse's Mouth* is risky while the double dice icon suggests that *Akahige* is very risky (McNee et al., 2003b).

it is very risky due to very few ratings in the database. Results indicated that such graphics were effective at conveying a system's confidence in its recommendations, but that minimal training was appropriate to teach users how to interpret these graphics.

More detailed methods of explaining how recommendations were generated can provide more information to a user to aid in her decision-making process. Four common models of explaining collaborative filtering recommendations are process, recommendation data, track record, and external data. Process explanations address the manner in which recommendations were produced, helping users determine whether that manner is appropriate for the decision being considered. Recommendation data can be explained generally, through summarization, or they can be presented or visualized in greater detail. The recommender system's track record for the user or in similar cases may help users gauge how well the system is likely to perform for them. Finally, external data, while not used in the recommendation process, can provide additional input to help users determine whether to follow a particular recommendation. These explanation models can be combined to create more elaborate recommendations. Additional explanation models can be used to address social navigations systems with data other than ratings.

Herlocker et al. (2000) studied 21 different explanations for Movie recommendations. Six of these explanations had a statistically significant positive effect on decision-making: three simple displays of neighbor ratings, one measure of historical correctness, and two directly content-focused explanations—the similarity of the recommended movie to other movies rated and the assertion that one of the user's favorite actors is in the cast. It should not be surprising that explanations that tie to content and semantics would be appealing to users—these explanations speak directly to user interests. A key conclusion is that Web recommender system effectiveness, overall, can be greatly enhanced by providing explanations that exploit any available semantic data. Similarly, automated markup systems may help users better cope with mistakes by offering users explanations for the markups made.

6.5.2 Focusing Implicit Ratings

Recommender system researchers have been studying ways in which ratings can be collected from users with the least effort and the most value. One promising approach is *implicit ratings*. Implicit ratings are measures of interest that are derived from observing the user's interaction with a site without asking the user to modify his behavior in any way. For instance, Tapestry stored annotations with documents identifying whether a user had printed or replied to a message (Goldberg et al., 1992). Other users could base their decisions on these annotations if they chose. The GroupLens project experimented with time-spent-reading as an implicit rating for Usenet news articles (Miller et al., 1997). Building on the work by Morita and Shinoda (1994) that showed that users spent more time reading articles they preferred, the GroupLens project was able to show that collaborative filtering was able to predict ratings for substantially more articles by using time-spent-reading and that the ratings predicted were of similar quality to explicit ratings-based predictions.

Another implicit rating is available in the link structure of the Web. Web links, in addition to providing direct access from one Web page to another, represent the Web page author's endorsement of the target page as useful and relevant, at least

Figure 6.6 A set of recommendations from PHOAKS. Each recommendation shows the number of distinct posters who mentioned the page, and provides links to their messages, a form of explanation (Terveen et al., 1997).

in context. Terveen and Hill (1998) found these data to be useful in classifying sites, specifically for identifying authoritative sites and information hubs. The Google search engine employs a similar technique to rank results returned from a search in an attempt to present the most important sites first; sites that are linked to from many other authoritative sites are considered more important than sites that are sparsely linked to. Hill and Terveen (1996) even found that cross-medium recommendations were effective, mining endorsements of Web pages from Usenet news articles in the PHOAKS system shown in Figure 6.6 (Terveen et al., 1997). Both Terveen and Hill's systems and Google use only information already created by authors for another purpose, benefiting from both the availability and the relative objectivity of implicit ratings.

These implicit ratings could be improved if the meaning of the pages or links was understood better. Did the author link to the page because it was an example of a bad page on a given topic? The Semantic Web might be able to record this information so recommenders could use it. Further, if a user frequently visits Web pages on many sites that have particular Semantic Web tags, that user may like Web pages on other sites that have those same tags. Typical implicit ratings systems would not detect the similarity between these pages, since the URLs are different. Combining the technology for extracting implicit ratings with the Semantic Web's rich data about each page might make possible richer implicit ratings systems that would be even more effective at extracting meaning for users.

6.6 Socially Aware Recommenders

Recommender systems started as systems that used group information to help each individual, but too often these systems ignore the importance of the groups themselves. This section discusses social navigation systems, which make community behavior visible in aggregate, and recommenders for smaller subgroups of a larger community.

6.6.1 Social Navigation

Social navigation systems have emerged as a broad array of techniques that enable people to work together to help each other find their way through crowded spaces. In a social navigation system, each user who visits a Web site does a small amount of work to untangle which of the paths from that Web site are most valuable. Early users leave information signposts that help later users make sense of the wealth of alternatives available to them. Later users benefit from the signposts, because they are able to direct their attention to the parts of the site that are most valuable to them. Some forms of social navigation are very closely related to collaborative filtering. For instance, Höök et al. (2003) discuss ways in which the passage of users through an information space can leave footprints that help other users find their way more readily through that same space (see Figure 6.7). As information spaces become more crowded with users, we may find it important to have automated systems that show us only those footprints that are most useful to us.

Figure 6.7 Footprints show Web visitors which links the community has followed.

Social navigation may fit into the Semantic Web in many different ways. Most directly, social navigation may extend to systems in which visitors to Web pages explicitly label those pages according to their semantic content. After all, labeling only by the authors of the pages is limiting because the authors are generally at a very different level in their understanding of the content than most visitors. As different visitors label pages, collaborative filtering can be used to select the view of the Semantic Web that is most valuable for each visitor, according to his viewing history and relationships with the previous visitors.

Social navigation can also fit into the Semantic Web by making visible which of the labels on a page seem most appropriate with respect to what visitors to that page actually do. For instance, one page might be labeled as having information on both sailing and windsurfing; however, if all visitors to that page end up following links to windsurfing information, social navigation could adjust the relative strengths of the labels to indicate that windsurfing is a more important label for that page.

Social navigation and the Semantic Web have similar goals. Both provide facilities that make navigating the Web more efficient and successful for users. The two techniques are complementary, and over time we anticipate that they will be used synergistically.

6.6.2 Recommending for Groups

Most automated collaborative filtering systems have focused exclusively on recommending items to individuals. In some domains, such as Usenet News (Resnick et al., 1994; Konstan et al., 1997; Miller et al., 1997), this limitation is understandable. Few users read articles collectively. In other domains such as books or music (Shardanand and Maes, 1995), it is common to enjoy the media both alone and in groups. Moreover certain items, among them board games and movies (Hill et al., 1995), are more commonly enjoyed in groups. Recommender systems that identify items such as movies for individuals do not address the user's key question, which is not "What movie should I see?" but rather "What movie should we see?" Recommendations can be provided to a group either by extending the automatic recommender to automatically recommend for the group, or by creating a new interface that displays individual recommendations for each member of the group, so the users can form their own group recommendation. In either case, the problem requires extending existing recommendation algorithms.

We built such a group recommendation interface for our MovieLens research Web site. We call the group interface PolyLens. PolyLens encourages users to form groups of other MovieLens users, using either the e-mail address or MovieLens pseudonym of the other users. Once a group is formed, each user in the group can get lists of recommended movies with the PolyLens prediction of how much that user will like the movie, and a separate prediction of how much the whole group will like the movie (see Figure 6.8). Overall, our experience has been that users are enthusiastic about group recommenders, and that group recommenders would help users make choices (O'Connor, 2001).

MusicFX is another system that supports group formation, though without using collaborative filtering (McCarthy and Anagnost, 1998). MusicFX was designed to address the challenge of selecting music for the often-large groups of people using a corporate gym. Each person filled out a taste profile describing his or her music preferences. The computer that tracked who was in the gym also kept track of what type of music would be most appropriate for the current group of people who were

Group: Dantest		Back To Individual Recommendations				
TITLE	GENRE	REVIEWS	GROUP	YOUR	cooley@cs.wmich.edu	cooley@quasar
Pixote (1987)	Drama		★★★★★	★★★★★	★★★★★	★★★★★
Young Tom Bruns, The (1993)	Adventure, Comedy		★★★★★	★★★★★	★★★★★	★★★★★
After Life (1994)	Drama		★★★★★	★★★★★	★★★★★	★★★★★
King of Masks: The (Black Lion) (1995)	Drama		★★★★★	★★★★★	★★★★★	★★★★★

Figure 6.8 PolyLens group recommendation display. Iterative user testing showed that users preferred a single list with separate recommendations for the group and for each individual group member.

working out. Over time, some people changed their work habits to arrange to be at the gym when other people, often strangers, with similar musical tastes were there. Figure 6.9 shows the system's view of five workout participants and their taste in music.

Currently most Web navigation is solo. Group navigation is likely to become increasingly important over time, much as multiplayer games are becoming prevalent, and multi-user support is now expected in document authoring systems. For instance, MultECommerce supports a group of shoppers visiting a virtual store together (Puglia et al., 2000). A recommender system in such a store should be integrated into the group experience, perhaps providing recommendations of items for individuals in the group, but basing the recommendations in part on the taste of the entire group. For instance, the recommendations of clothing shown to a group of teenage girls shopping alone might be very different to the clothing recommendations shown to one of the girls shopping with her mother.

Similarly, Maglio et al. (2003) combine the social navigation from the prior section with group formation as a way to make the Web a more social space. They propose a means of connecting individuals as they traverse through the Web. In essence, they

<i>i</i>	Genre	Person	A	B	C	D	E	GP _i	Pr _i
1	Alternative Rock		2	2	0	2	2	68	0.48
2	Hottest Hits		1	1	2	0	-2	38	0.27
3	New Music		1	1	1	0	0	35	0.25
4	Hot Country		2	0	0	0	-2	28	0.00
5	Dance		2	-1	1	-1	-1	28	0.00
6	World Beat		0	1	-1	1	-2	23	0.00
7	Traditional Country		1	0	0	-2	-2	17	0.00
8	60's Oldies		0	0	0	-1	-1	14	0.00
9	Heavy Metal		-1	-1	-1	-1	-2	4	0.00
10	Polka		-1	-1	-2	-2	-2	2	0.00

Figure 6.9 Example preferences from the MusicFX system. Users A–E are the five users currently in the gym. Each of the users A–E has specified her preference for each radio station that could be chosen to be played in the gym. MusicFX has computed an aggregate preference score for each of the radio stations for the five users. The stations are sorted by aggregate preferences. In this case, the top three stations dominate for these five users, so those stations will be played most of the time. Note that MusicFX is not a winner-takes-all system: stations with lower aggregate preference will still be played, though less frequently. (From McCarthy and Anagnost, 1998.)

create a recommender whose recommendations are other individuals who share the same “place” on the Web as the user receiving the recommendations. They discuss a variety of ways to define what it means to share the same place, from the trivial (people sharing the same host organization) to the more complex (people who are browsing the Web within several links of each other). Their belief is that knowledge of such individuals may make the Web a more social place to be.

Ludford et al. (2004) also consider group formation but extend this to include a mechanism to provide for continued group involvement. Historically, and as a whole, online groups have a tendency to go dormant and over time die out. Ludford et al. propose that one reason for this lack of activity is the failure for online groups to generate a proper social structure. Results of their controlled studies in the domain of movies indicate that groups formed from individuals with a dissimilar set of interests actually generated more activity than those formed from similar individuals. Furthermore, they discovered that individuals who understood what made them unique to the community (or at the very least, to the current discussion) would often exploit their uniqueness to contribute to the community or spark additional discussion. Perhaps these results can be used by designers of recommenders to create sites and communities that encourage contributions to the system, and thus, the betterment of the community.

Similar to the changes in recommender interfaces that have been necessary to support groups, the interface to the Semantic Web will have to change to support groups. One such interface might be a recommender supporting a group of people navigating together, and using both collaborative filtering and Semantic Web information to make its recommendations. Such an interface might provide a view of the Semantic Web that would be appropriate to the group of individuals navigating together. Each member of the group might see some pages that are selected to be appropriate for just him, a group view that is appropriate for the group together, and see or hear comments made by group members about the pages. So little work has been done in this area that it is hard to know which interface will be most successful, but experience suggests that making navigating the Web a social experience will be important over the long term.

6.7 Portable Recommenders

A key advantage of recommenders is their helpfulness in suggesting a select subset of available information to be displayed in a limited amount of screen real estate. This advantage of recommenders is even more valuable in small devices, on which the amount of screen real estate is minuscule, and through which users want immediately valuable information without having to browse through a large number of alternatives.

Miller et al. (2003) examined four interfaces to recommendations that could be used on mobile devices: (1) WAP, a Web technology for cell phone screens; (2) Avant-Go, a technology for synchronizing selected Web pages to a PDA; (3) live HTML on a PDA; and (4) a VoiceXML interface for speaking to the recommender over a telephone. The overall results were that users very much liked the portable recommenders, because they could get recommendations right when and where they needed them. On the bad side, each of the four interfaces required significant design particular to the device and interface technology used. A generalized framework for developing small-device information browsing interfaces would be invaluable, though we seem many years of research away from such a framework. Users also found significant advantages to the

Internet-connected recommenders, because those interfaces had all of the information available, and had the most up-to-date information. However, users of these interfaces found connectivity frustrating, since wide-area wireless Internet networks are far from ubiquitous and reliable.

One solution to the problem of connectivity is to create a recommender algorithm that can run directly on the small device. Miller et al. (2004) explored recommender algorithms that could compute recommendations on a PDA-sized computer in real-time. They were able to show that such algorithms can produce accuracy nearly as good as the best workstation-based recommenders, with a small memory footprint, and fast interactive performance. The key insight is to specialize the model stored on the PDA for the single user of that PDA. The authors postulate that such a recommender frees a user from the influence of the operator of a centralized recommender, who may have goals (such as selling overstock products) that are not shared by the user.

Access to information on small devices seems likely to be increasingly important. Recommenders can be a valuable part of small-device information browsers, because they can dramatically reduce the amount of information the user must sift through to find the items he is interested in. Developing such recommender algorithms on large computers is relatively straightforward with modern technology, and the resulting interfaces can be displayed on the portable devices. If the device is not always connected to the Internet when information is desired, it may be possible to develop new algorithms that can perform the recommender tasks on the small device.

6.8 Cheating with Recommenders

One important question in recommender systems is the extent to which users can manipulate the system by putting in ratings that are not representative of their real preferences. For instance, a record label might pay a thousand users to falsely claim to love a newly released single. There are many examples of this sort of manipulation in real-world recommenders (e.g., Sony using fake quotes from a made-up film critic to promote its films (BBC News, 2001)), so it seems likely to happen with online recommenders, too. Several empirical studies have explored the effectiveness of putative attacks on recommender systems. O'Mahoney et al. (2003) showed that attacks on kNN users-user algorithms could both *push* items by raising predicted ratings, and *nuke* items by lowering predicted ratings. Lam et al. [2] demonstrated that the kNN item-item algorithm seems somewhat less susceptible to these attacks—though this paper also showed that the methods recommender systems researchers use to look for these attacks are not sensitive even to very harsh attacks on the system. Much more research is needed to understand how to identify attacks when they occur, and—crucially—how to protect recommender systems from such attacks. Similar attacks are likely on semantic Web sites that use information from the community to label the sites.

6.9 Conclusion

Recommender systems already provide substantial user value by personalizing a number of sites on the Web. The Semantic Web brings forward rich opportunities for improving these interfaces, and for striking a better balance between content and collaborative personalization methods.

6.10 Acknowledgments

We gratefully acknowledge the contributions of the students who are current and alumni members of the GroupLens Research Group at the University of Minnesota, without whom this work could not have been done. This work was supported by the National Science Foundation under grants IIS 9613960, IIS 9734442, IIS 9978717, and DGE 9554517. Support was also provided by Net Perceptions Inc., a company that we co-founded that sells an automatic collaborative filtering system.

6.11 References

- Balabanovic, M., Shoham, Y. (1997). Fab: Content-based, collaborative recommendation. *Communications of the ACM*, 40(3).
- BBC News (2001). *Sony Admits to Using Fake Reviewer*. Retrieved from <http://news.bbc.co.uk/1/hi/entertainment/film/1368666.stm>.
- Bharat, K., Kamba, T., Albers, M. (1998). Personalized, interactive news on the Web. *Multimedia Systems*, 6(5):349–358.
- Breese, J., Heckerman, D., Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI-98)*, Morgan Kaufmann, pp. 43–52.
- Burke, R. (2002). Hybrid recommender systems: Survey and experiments. In: *User Modeling and User-Adapted Interaction*, 12:331–370, Kluwer Academic Publishers.
- Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D., Sartin, M. (1999). Combining content-based and collaborative filters in an online newspaper. *Proceedings of the SIGIR 1999 Workshop on Recommender Systems: Algorithms and Evaluation*. Available: <http://www.cs.umbc.edu/~ian/sigir99-rec/>.
- Cosley, D., Lam, S.K., Albert, I., Konstan, J.A., Riedl, J. (2003). Is seeing believing? How recommender systems influence users' opinions. *Proceedings of CHI 2003 Conference on Human Factors in Computing Systems*, Fort Lauderdale, FL, pp. 585–592.
- Cranor, L. (2002). *Web Privacy with P3P*. O'Reilly & Associates.
- Deshpande, M., Karypis, G. (2004). Item-based Top-N recommendation algorithms. *ACM Transactions on Information Systems*, 22(1):143–177.
- Goldberg, D., Nichols, D., Oki, B.M., Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12):61–70.
- Good, N., Schafer, J.B., Konstan, J.A., Borchers, A., Sarwar, B., Herlocker, J., Riedl, J. (1999). Combining collaborative filtering with personal agents for better recommendations. *Proceedings of AAAI-99*, AAAI Press, pp. 439–446.
- Herlocker, J., Konstan, J.A. (2001). Content-independent task-focused recommendation. *IEEE Internet Computing*, 5(6).
- Herlocker, J., Konstan, J.A., Borchers, A., Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. *Proceedings of SIGIR'99*, ACM, pp. 230–237.
- Herlocker, J., Konstan, J.A., Riedl, J. (2000). Explaining collaborative filtering recommendations. *Proceedings of the ACM 2000 Conference on Computer-Supported Cooperative Work*, ACM, pp. 241–250.
- Herlocker, J., Konstan, J.A., Terveen, L., Riedl, J. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1):5–53.
- Hill, W., Terveen, L. (1996). Using frequency-of-mention in public conversations for social filtering. *Proceedings of the ACM 1996 Conference on Computer-Supported Cooperative Work*, ACM, pp. 106–112.
- Hill, W., Stead, L., Rosenstein, M., Furnas, G. (1995). Recommending and evaluating choices in a virtual community of use. *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, ACM, pp. 194–201.
- Hofmann, T. (2004). Latent semantic models for collaborative filtering. *ACM Transactions on Information Systems*, 22(1):89–115.
- Höök, K., Benyon, D., Munro, A. (2003). Footprints in the snow. In: Höök, K., Benyon, D., Munro, A. (Eds.), *Social Navigation of Information Space*. Springer-Verlag, London.
- Jin, R., Si, L., Zhai, C., Callan, J. (2003). Collaborative filtering with decoupled models for preferences and ratings. *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, ACM, pp. 309–316.

- Konstan, J.A., Miller, B., Maltz, D., Herlocker, J., Gordon, L., Riedl, J. (1997). GroupLens: Applying collaborative filtering to Usenet news. *Communications of the ACM*, 40(3):77–87.
- Lam, S.K., Riedl, J. (2004). Shilling Recommender systems for fun and profit. *Proceedings of WWW 2001*, WWW10 Limited, pp. 393–394.
- Ludford, P., Cosley, D., Frankowski, D., Terveen, L. (2004). Think different: Increasing online community participation using uniqueness and group dissimilarity. *Proceedings of the 2004 Conference on Human Factors in Computing Systems*, ACM, pp. 631–638.
- Maglio, P., Barrett, R., Farrell, S. (2003). WebPlaces: Using intermediaries to add people to the Web. In: Höök, K., Benyon, D., Munro, A. (Eds.), *Social Navigation of Information Spaces*. Springer-Verlag, London.
- O'Mahony, M.P., Hurley, N., Kushmerick, N., Silvestre, G. (2003). Collaborative recommendation: A robustness analysis. *ACM Transactions on Internet Technology*, 2003. Special Issue on Machine Learning for the Internet, 4(3).
- Maltz, D., Ehrlich, E. (1995). Pointing the way: Active collaborative filtering. *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, ACM, pp. 202–209.
- McCarthy, J., Anagnost, T. (1998). MusicFX: An arbiter of group preferences for computer supported collaborative workouts. *Proceedings of the ACM 1998 Conference on Computer Supported Cooperative Work*, ACM, pp. 363–372.
- McLaughlin, M., Herlocker, J. (2004). A collaborative filtering algorithm and evaluation metric that accurately model the user experience. *Proceedings of the ACM SIGIR'04 Conference*, ACM, pp. 329–336.
- McNee, S., Albert, I., Cosley, D., Gopalkrishnan, P., Lam, S.K., Rashid, A.M., Konstan, J.A., Riedl, J. (2002). On the recommending of citations for research papers. *Proceedings of 2002 Conference on Computer Supported Cooperative Work*, ACM, pp. 116–125.
- McNee, S., Lam, S.K., Konstan, J.A., Riedl, J. (2003a). Interfaces for eliciting new user preferences in recommender systems. *Proceedings of the 9th International Conference on User Modeling (UM'2003)*, LNAI 2702, pp. 178–188.
- McNee, S., Lam, S.K., Guetzlaff, C., Konstan, J.A., Riedl, J. (2003b). Confidence displays and training in recommender systems. *Proceedings of INTERACT '03 IFIP TC13 International Conference on Human-Computer Interaction*, pp. 176–183.
- Middleton, S., Shadbolt, N., De Roure, D. (2004). Ontological user profiling in recommender systems. *ACM Transactions on Information Systems*, 22(1):54–88.
- Miller, B., Riedl, J., Konstan, J.A. (1997). Experiences with GroupLens: Making Usenet useful again. *Proceedings of the 1997 Usenix Winter Technical Conference*, USENIX, January 1997.
- Miller, B., Albert, I., Lam, S.K., Konstan, J.A., Riedl, J. (2003). MovieLens unplugged: Experiences with a recommender system on four mobile devices. *Proceedings of the 17th Annual Human-Computer Interaction Conference (HCI 2003)*, British HCI Group, September 2003.
- Miller, B., Konstan, J.A., Terveen, L., Riedl, J. (2004). PocketLens: Towards a personal recommender system. *ACM Transactions on Information Systems*, 22(3):437–476, July 2004.
- Morita, M., Shinoda, Y. (1994). Information filtering based on user behavior analysis and best match text retrieval. *Proceedings of the 17th Annual International SIGIR Conference on Research and Development*, ACM, pp. 272–281.
- O'Connor, M., Cosley, D., Konstan, J.A., Riedl, J. (2001). PolyLens: A recommender system for groups of users. *Proceedings of ECSCW 2001*, Kluwer Academic Publishers, Bonn, Germany.
- Puglia, S., Carter, R., Jain, R. (2000). MultiECommerce: A distributed architecture for collaborative shopping on the WWW. *Proceedings of the 2nd ACM Conference on Electronic Commerce*, ACM, Minneapolis, MN, pp. 215–224.
- Rashid, A.M., Albert, I., Cosley, D., Lam, S.K., McNee, S., Konstan, J.A., Riedl, J. (2002). Getting to know you: Learning new user preferences in recommender systems. *Proceedings of the 2002 International Conference on Intelligent User Interfaces*, ACM, pp. 127–134.
- Resnick, P., Miller, J. (1996). PICS: Internet access controls without censorship. *Communications of the ACM*, 39(10):87–93.
- Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J. (1994). Grouplens: An open architecture for collaborative filtering of netnews. *Proceedings of ACM CSCW'94 Conference on Computer-Supported Cooperative Work*, ACM, pp. 175–186.
- Sarwar, B., Konstan, J.A., Borchers, A., Herlocker, J., Miller, B., Riedl, J. (1998). Using filtering agents to improve prediction quality in the grouplens research collaborative filtering system. *Proceedings of 1998 Conference on Computer-Supported Collaborative Work*, ACM.
- Sarwar, B., Karypis, G., Konstan, J.A., Riedl, J. (2000). Analysis of recommender algorithms for e-commerce. *Proceedings of the ACM E-Commerce 2000 Conference*, ACM.
- Sarwar, B., Karypis, G., Konstan, J.A., Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. *Proceedings of WWW 2001*, WWW10 Limited.

- Schafer, J.B., Konstan, J.A., Riedl, J. (2001). Electronic commerce recommender applications. *Journal of Data Mining and Knowledge Discovery*, 5(1/2):115–152.
- Schafer, J.B., Konstan, J.A., Riedl, J. (2002). Meta-recommender systems: User-controlled integration of diverse recommendations. *Proceedings of the Eleventh International Conference on Information and Knowledge Management*, ACM, pp. 43–51.
- Schafer, J.B., Konstan, J.A., Riedl, J. (2005). The view through MetaLens: Usage patterns for a meta-recommender system. *IEEE Proceedings Software*, IEE.
- Shardanand, U., Maes, P. (1995). Social information filtering: Algorithms for automating “word of mouth.” *Proceedings of ACM CHI’95 Conference on Human Factors in Computing Systems*, ACM, pp. 210–217.
- Terveen, L., Hill, W. (1998). Finding and visualizing inter-site clan graphs. *Proceedings of ACM CHI 98 Conference on Human Factors in Computing Systems*, ACM, pp. 448–455.
- Terveen, L., Hill, W., Amento, B., McDonald, D., Creter, J. (1997). PHOAKS: A system for sharing recommendations. *Communications of the ACM*, 40(3):59–62.
- Torres, R., McNee, S., Abel, M., Konstan, J.A., Riedl, J. (2004). Enhancing digital libraries with TechLens. *Proceedings of the 2004 Joint ACM/IEEE Conference on Digital Libraries*, ACM, pp. 228–236.
- Wolf, J., Aggarwal, C., Wu, K-L, Yu, P. (1999). Horting hatches an egg: A new graph-theoretic approach to collaborative filtering. *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, San Diego, CA.