Data Mining and Knowledge Discovery, 5, 115–153, 2001

E-Commerce Recommendation Applications

J. BEN SCHAFER* schafer@cs.umn.edu
JOSEPH A. KONSTAN konstan@cs.umn.edu
JOHN RIEDL riedl@cs.umn.edu
GroupLens Research Project, Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455, USA

Editors: Ron Kohavi and Foster Provost

Abstract. *Recommender systems* are being used by an ever-increasing number of E-commerce sites to help consumers find products to purchase. What started as a novelty has turned into a serious business tool. Recommender systems use product knowledge—either hand-coded knowledge provided by experts or "mined" knowledge learned from the behavior of consumers—to guide consumers through the often-overwhelming task of locating products they will like. In this article we present an explanation of how recommender systems are related to some traditional database analysis techniques. We examine how recommender systems help E-commerce sites increase sales and analyze the recommender systems at six market-leading sites. Based on these examples, we create a taxonomy of recommender systems, including the inputs required from the consumers, the additional knowledge required from the database, the ways the recommendations are presented to consumers, the technologies used to create the recommendations, and the level of personalization of the recommendations. We identify five commonly used E-commerce recommender application models, describe several open research problems in the field of recommender systems, and examine privacy implications of recommender systems technology.

Keywords: electronic commerce, recommender systems, personalization, customer loyalty, cross-sell, up-sell, mass customization, privacy, data mining, database marketing, user interface

1. Introduction

> "If I have 3 million customers on the Web, I should have 3 million stores on the Web."
>
> —Jeff Bezos, CEO of Amazon.com™

Imagine a physical world where there are hundreds upon hundreds of branches of the same store. I have a branch in my neighborhood tailored to my needs, and you have a branch in your neighborhood tailored to yours. In the physical world this would be impossible (notwithstanding a Starbucks on every corner); however, the movement toward E-commerce—commerce in the virtual space—has produced business strategies that could never exist in the physical world. In his book *Mass Customization* (Pine, 1993), Joe Pine argues that companies need to shift from the old world of mass production where "standardized products, homogeneous markets, and long product life and development cycles were the rule" to the

*To whom correspondence should be addressed at 4-192 EE/CSci Building, 200 Union Street S.E., University of Minnesota, Minneapolis, Minnesota 55455, USA.

new world where “variety and customization supplant standardized products.” Pine argues that building one product is simply not adequate anymore. At a minimum, companies need to be able to develop *multiple* products that meet the *multiple* needs of *multiple* consumers. While E-commerce hasn’t necessarily allowed businesses to produce more products, it has allowed them to provide consumers with more choices. Instead of tens of thousands of books in a superstore, consumers may choose among millions of books in an online store. Increasing choice, however, has also increased the amount of information that consumers must process before they are able to select which items meet their needs. To address this information overload, e-commerce stores are applying mass customization principles not to the products but to their presentation in the on-line store (Pine and Gilmore, 1999). One way to achieve mass customization in e-commerce is the use of *recommender systems*.

Recommender systems[1] are used by E-commerce sites to suggest products to their customers and to provide consumers with information to help them decide which products to purchase. The products can be recommended based on the top overall sellers on a site, on the demographics of the consumer, or on an analysis of the past buying behavior of the consumer as a prediction for future buying behavior. The forms of recommendation include suggesting products to the consumer, providing personalized product information, summarizing community opinion, and providing community critiques. Broadly, these recommendation techniques are part of personalization on a site because they help the site adapt itself to each customer. Personalization, to this extent, is one way to realize Pine’s ideas on the Web. Mass customization originally referred to the physical modification of products and services to make them fit each consumer’s needs (Pine, 1993). More recently, mass customization has evolved to encompass a wide range of methods for customizing the *consumer experience* (Pine and Gilmore, 1999). The consumer experience includes the physical products, which can be customized in function or in appearance, and the presentation of those products, which can be customized automatically or with help from the consumer. Under this broader definition, recommender systems serve to support a customization of the consumer experience in the presentation of the products sold on a Web site. In a sense, recommender systems enable the creation of a new store personally designed for each consumer. Of course, in the virtual world, all that changes is the selection of products shown to the consumer, not an underlying physical store.

Recommender systems are similar to, but also different from, marketing systems and supply-chain decision-support systems. Marketing systems support the marketer in making decisions about how to market products to consumers, usually by grouping the consumers according to marketing segments and grouping the products in categories that can be aligned with the marketing segments. Marketing campaigns can then be run to encourage consumers in different segments to purchase products from categories selected by the marketer. By contrast, recommender systems directly interact with consumers, helping them find products they will like to purchase. Supply-chain decision-support systems help marketers make decisions about how many products to manufacture, and to which warehouses or retail stores to ship the products. These decision-support systems use analytic technology to predict how many of which products will be purchased in each location, so the right products are available for consumers to purchase. Many supply-chain decision-support systems answer questions about aggregates: of all the consumers in Minneapolis, how many will buy toothpaste in

February? Recommender systems answer questions about individual consumers: which product will this consumer prefer to buy right now?

Recommender systems include processes that are conducted largely by hand, such as manually creating cross-sell lists, and actions that are performed largely by computer, such as collaborative filtering. We will refer to the latter as *automatic recommender systems*. Automatic recommender systems are specialized data mining systems that have been optimized for interaction with consumers rather than marketers. They have been explicitly designed to take advantage of the real-time personalization opportunities of interactive e-commerce. Accordingly, the algorithms focus more on real-time and just-in-time learning than on model-building and execution. We study both manual and automatic recommender systems since each offers many interesting ideas about the presentation of recommendations to consumers. This paper serves as an introduction to the elements of recommender systems and their application to e-commerce.

Recommender systems enhance E-commerce sales in three ways:

Converting Browsers into Buyers: Visitors to a Web site often look over the site without purchasing anything. Recommender systems can help consumers find products they wish to purchase.

Increasing Cross-sell: Recommender systems improve cross-sell by suggesting additional products for the customer to purchase. If the recommendations are good, the average order size should increase. For instance, a site might recommend additional products in the checkout process, based on those products already in the shopping cart.

Building Loyalty: In a world where a site's competitors are only a click or two away, gaining consumer loyalty is an essential business strategy (Reichheld and Sesser, 1990; Reichheld, 1993). Recommender systems improve loyalty by creating a value-added relationship between the site and the customer. Sites invest in learning about their customers, use recommender systems to operationalize that learning, and present custom interfaces that match consumer needs. Consumers repay these sites by returning to the ones that best match their needs. The more a customer uses the recommendation system—teaching it what he wants—the more loyal he is to the site. "Even if a competitor were to build the exact same capabilities, a customer . . . would have to spend an inordinate amount of time and energy teaching the competitor what the company already knows" (Pine et al., 1995). Creating relationships between consumers can also increase loyalty, for consumers will return to the site that recommends people with whom they will like to interact.

This paper makes five contributions to the understanding of the application of recommender systems in E-commerce. First, we examine how traditional marketing methods provided a foundation for the growth of recommender systems as a marketing tool in E-commerce. Second, we present a taxonomy for Recommender Applications, classifying them based on the inputs to the recommender process, the method used to generate recommendations, the outputs of the recommendation process to the customer, and the degree of personalization. Third, we examine the patterns that emerge when considering the taxonomy and identify five models of recommender applications. These five models are currently the dominant uses of recommender systems in E-commerce. Fourth, we describe four domains of future study for new recommender system applications based on parts of our taxonomy that have not been adequately explored by the existing applications. Finally, in the appendix, we consider privacy issues that are evolving as more sites begin to implement recommender applications.

The paper is useful to two groups: academics studying recommender systems in E-commerce and implementers considering applying recommender systems in their site. For academics, the examples and taxonomies provide a useful initial framework within which their research can be placed. The framework will undoubtedly be expanded to include future applications of recommender systems. Also, the paper identifies research challenges in recommender systems for the data mining community. For implementers, the paper provides a means of making choices among the available applications and technologies. An implementer can choose a moneymaking goal, select the interfaces that will help achieve that goal, and pick an implementation technique that supports the goal within the interface.

This paper differs from our earlier work (Schafer et al., 1999) in several key ways. First, the examples have been updated and expanded to better reflect the rapidly expanding field of recommender systems. Second, the taxonomy has been modified and expanded to more accurately encompass all of the aspects of recommendation technology and to be appropriate for a data mining audience. Third, the opportunities section has been expanded to feature additional ideas and to reflect the current state of the field. Finally, several new sections have been added, including sections relating recommender systems to traditional marketing techniques and a discussion of privacy concerns.

2. Prior and related work

As merchandisers gained the ability to record transaction data, they started collecting and analyzing data about consumer behavior. The term *data mining* is used to describe the collection of analysis techniques used to infer rules from or build models from large data sets. One of the best-known examples of data mining in commerce is the discovery of association rules—relationships between items that indicate a relationship between the purchase of one item and the purchase of another. These rules can help a merchandiser arrange products so that, for example, a consumer purchasing ketchup sees relish nearby. More sophisticated temporal data mining may suggest that a consumer who buys a new charcoal grill today is likely to buy a fire extinguisher in the next month.

More generally, data mining has two phases. In the learning phase, the data mining system analyzes the data and builds a model of consumer behavior (e.g., association rules). This phase is often very time-consuming and may require the assistance of human analysts. After the model is built, the system enters a use phase where the model can be rapidly and easily applied to consumer situations. One of the challenges in implementing data mining within organizations is creating the organizational processes that successfully transfer the knowledge from the learning phase into practice in the use phase.

Automatic recommender systems are machine learning systems specialized to recommend products in commerce applications. Some recommenders have an offline phase during which they learn a model of customer behavior, and then an online phase during which they apply the model in real time. Most recommenders, however, use a lazy learning approach, in which they build and update the model while making recommendations in real time.

Approaches

Many different approaches have been applied to the basic problem of making accurate and efficient recommender and data mining systems. Many of the technologies used in the actual recommender systems studied are fairly simple database queries. Automatic recommender systems, however, use a wide range of techniques, ranging from nearest neighbor algorithms to Bayesian analysis. The worst-case performance of many of these algorithms is known to be poor. However, many of the algorithms have been tuned to use heuristics that are particularly efficient on the types of data that occur in practice.

The earliest recommenders used nearest-neighbor collaborative filtering algorithms (Resnick et al., 1994; Shardanand et al., 1995). Nearest neighbor algorithms are based on computing the distance between consumers based on their preference history. Predictions of how much a consumer will like a product are computed by taking the weighted average of the opinions of a set of nearest neighbors for that product. Neighbors who have expressed no opinion on the product in question are ignored. Opinions should be scaled to adjust for differences in ratings tendencies between users (Herlocker et al., 1999). Nearest neighbor algorithms have the advantage of being able to rapidly incorporate the most up-to-date information, but the search for neighbors is slow in large databases. Practical algorithms use heuristics to search for good neighbors and may use opportunistic sampling when faced with very large populations.

Bayesian networks create a model based on a training set with a decision tree at each node and edges representing consumer information. The model can be built off-line over a matter of hours or days. The resulting model is very small, very fast, and essentially as accurate as nearest neighbor methods (Breese et al., 1998). Bayesian networks may prove practical for environments in which knowledge of consumer preferences changes slowly with respect to the time needed to build the model but are not suitable for environments in which consumer preference models must be updated rapidly or frequently.

Clustering techniques work by identifying groups of consumers who appear to have similar preferences. Once the clusters are created, predictions for an individual can be made by averaging the opinions of the other consumers in that cluster. Some clustering techniques represent each consumer with partial participation in several clusters. The prediction is then an average across the clusters, weighted by degree of participation. Clustering techniques usually produce less-personal recommendations than other methods, and in some cases, the clusters have worse accuracy than nearest neighbor algorithms (Breese et al., 1998). Once the clustering is complete, however, performance can be very good, since the size of the group that must be analyzed is much smaller. Clustering techniques can also be applied as a “first step” for shrinking the candidate set in a nearest neighbor algorithm or for distributing nearest-neighbor computation across several recommender engines. While dividing the population into clusters may hurt the accuracy or recommendations to users near the fringes of their assigned cluster, pre-clustering may be a worthwhile trade-off between accuracy and throughput.

Information filtering and *information retrieval* involve selecting text items that a user may be interested in reading based on the presence or absence of keywords in the text items. The user can explicitly enter the keywords, or they can be inferred from the items that the user has found interesting in the past. Information filtering or information retrieval systems are often used in search systems on e-commerce sites to help consumers find specific products in which they are interested. These systems have some features in common with recommender systems, in that both systems produce lists of suggestions for a user; however, the more the system provides direct responses to syntactic user queries the less it feels like a recommender system to the user. Information filtering systems that notify users when interesting items are for sale are more like recommender systems, especially if part of the selection process involves attributes that are not under the user's direct control, such as whether other users have liked the item.

Classifiers are general computational models for assigning a category to an input. The inputs may be vectors of features for the items being classified or data about relationships among the items. The category is a domain-specific classification such as malignant/benign for tumor classification, approve/reject for credit requests, or intruder/authorized for security checks. One way to build a recommender system using a classifier is to use information about a product and a customer as the input, and to have the output category represent how strongly to recommend the product to the customer. Classifiers may be implemented using many different machine-learning strategies including rule induction, neural networks, and Bayesian networks. In each case, the classifier is trained using a training set in which ground truth classifications are available. It can then be applied to classify new items for which the ground truths are not available. If subsequent ground truths become available, the classifier may be retrained over time.

Classifiers have been quite successful in a variety of domains ranging from the identification of fraud and credit risks in financial transactions to medical diagnosis to intrusion detection. Basu et al. (1998) built a hybrid recommender system that mixes collaborative and content filtering using an induction-learning classifier. Good et al. (1999) implemented induction-learned feature-vector classification of movies and compared the classification with nearest-neighbor recommendation; this study found that the classifiers did not perform as well as nearest neighbor, but that combining the two added value over nearest-neighbor alone.

Association rules have been used for many years in merchandising, both to analyze patterns of preference across products, and to recommend products to consumers based on other products they have selected. An association rule expresses the relationship that one product is often purchased along with other products. The number of possible association rules grows exponentially with the number of products in a rule, but constraints on confidence and support, combined with algorithms that build association rules with itemsets of n items from rules with n-1 item itemsets, reduce the effective search space. Association rules can form a very compact representation of preference data that may improve efficiency of storage as well as performance. They are more commonly used for larger populations rather than for individual consumers, and they, like other learning methods that first build and then apply models, are less suitable for applications where knowledge of preferences changes rapidly. Association rules have been particularly successfully in broad applications such as shelf layout in retail stores. By contrast, recommender systems based on nearest neighbor techniques are easier to implement for personal recommendation in a domain where consumer opinions are frequently added, such as on-line retail.

Horting is a graph-based technique in which nodes are consumers, and edges between nodes indicate degree of similarity between two consumers (Wolf et al., 1999). Predictions are produced by walking the graph to nearby nodes and combining the opinions of the nearby consumers. Horting differs from nearest neighbor as the graph may be walked through other consumers who have not rated the product in question, thus exploring transitive relationships that nearest neighbor algorithms do not consider. In one study using synthetic data, Horting produced better predictions than a nearest neighbor algorithm (Wolf et al., 1999).

In this paper we review existing e-commerce implementations according to how they are presented to consumers. Most of the Web stores we review consider the algorithms they use to be proprietary. Many of these algorithms could be used while still presenting the same interface to the user. For this reason, our taxonomy is based on the basic approach to recommendation, rather than the specific technology used.

Marketing technologies

Recommender systems evolved in response to an increasing set of choices in products to buy and information to consume, combined with consumer frustration at a decreasing level of professional support for making these choices (i.e., fewer expert shopkeepers). These conditions created challenges for both consumers and merchandisers. Consumers experienced information overload and sought help in selecting from an overwhelming array of products while merchandisers lost their relationships with consumers and sought to re-build and deepen those relationships by better helping consumers find products of interest.

Recommender systems responded directly to consumers, giving them independent advice modeled after informal "word of mouth." At the same time, new database marketing techniques, data mining, and targeted advertising responded to merchandisers, giving them tools to respond to consumer needs, understand consumer behavior, and best use the limited available customer attention. This section briefly describes database marketing and targeted advertising technologies and their relationship to recommender systems.

Database marketing is an attempt by businesses to provide more personal service to their customers. Neighborhood shopkeepers knew their regular customers and could provide each one with personal assistance, services, and advice. Many businesses today cannot maintain that one-to-one human relationship because of the prevalence of much larger retail stores, low employee-to-customer ratios, and high turnover among employees. Some businesses responded by treating all consumers the same. Others used database marketing to divide consumers into segments based on demographic characteristics such as ZIP code, income, and occupation and marketed to each segment as a group. In some cases database marketing chiefly treats consumers according to their individual needs, but in other cases consumers treated as part of a segment find that the business no longer understands their individual preferences, needs, or desires.

One-to-one marketing (Peppers and Rogers, 1997) attempts to overcome the impersonal nature of marketing by using technology to assist businesses in treating each consumer individually. Part of one-to-one marketing is the capture and use of consumer preferences (e.g., learning that a particular customer always wants gifts shipped overnight or that a particular customer collects an entire line of porcelain dolls). Other parts involve changing business practices to use the consumer knowledge gathered by the business.

Recommender systems are a technology that helps merchandisers implement a one-to-one marketing strategy. The recommender system analyzes a database of consumer preferences to overcome the limitations of segment-based mass marketing by presenting each customer with a personal set of recommendations. Of course, recommender systems are not a complete solution. It is still necessary to record and use other consumer data, such as preferred credit card and shipping address, to deliver complete one-to-one service to consumers.

Ad targeting, or more *generally offer targeting*, is an attempt to identify which consumers should be made an offer based upon their prior behavior. Traditional marketers watch for a given "event" in a customer's life and then aim specific advertisements or offers to the consumer. When a consumer applies for his first credit card, he begins receiving offers from numerous banks for their version of the card. When he purchases a house, he begins receiving offers for loan consolidation, second mortgages, life insurance and aluminum siding. When he has a child, he finds himself inundated with advertisements for everything from diapers and formula to book clubs and, once again, life insurance.

Offer targeting treats consumers as both individuals and as members of a market group. Offers are typically made to all consumers whose names appear on a list (i.e., the "just acquired a mortgage" list). However, individual customers are added and removed from these lists based on their individual behavior. Achieving a "life event" gets a customer added to a list. Consumers who continue to ignore the offers will eventually be removed from the list.

Recommender systems are a technology that can help businesses decide to whom to make an offer. Such systems allow search engines and advertising companies to suggest advertisements or offers to display based on consumer behavior. Yahoo or Excite could use a recommender system to identify which banner ad to display based on which keywords the consumer queried, or to which subsection of the hierarchy a customer navigated. Not surprisingly, customers who enter the keywords "Buick Century" in a search engine may find a banner advertising the latest Buick product. Likewise, consumers searching through the NFL section at Yahoo may receive a banner add for SportsAuthority.com, while consumers navigating to the directory of insurance agents in Utah may find an ad for AccuQuote.

3. Recommender system examples

In the following section we present six E-commerce businesses that use one or more variations of recommender system technology in their web sites. For each site, and each variation, we give a brief description of the features of the system. In later sections we refer to these examples as we explain the types of recommendations provided, the type of technology used, and the types of information gathered. For organizational purposes these sites have been alphabetized. The descriptions of these sites are accurate as of this writing, though E-commerce applications of recommender systems are changing rapidly.

3.1. Amazon.com

We will focus here on recommender systems in the *book* section of Amazon.com.

Customers Who Bought: Like many E-commerce sites, Amazon.com™ (www.amazon.com) is structured with an information page for each book, giving details of the text and purchase information. The Customers Who Bought feature is found on the information page for each book in their catalog. It is in fact two separate recommendation lists. The first recommends books frequently purchased by customers who purchased the selected book. The second recommends authors whose books are frequently purchased by customers who purchased works by the author of the selected book.

Your Recommendations: Amazon also encourages direct feedback from customers about books they have read. Customers rate books they have read on a 5-point scale from "hated it" to "loved it." After rating a sample of books, customers may request recommendations for books that they might like. At that point, a half dozen non-rated texts are presented that correlate with the user's indicated tastes. Figure 1 shows a sample screen from Your Recommendations.

Eyes: The Eyes feature allows customers to be notified via email of new items that have been added to the Amazon.com catalog. Customers enter requests based upon author, title, subject, ISBN, or publication date information. Customers can use both simple and more complex Boolean-based criteria (AND/OR) for notification queries. One of the interesting variations of the Eyes system allows requests to be directly entered from any search results screen, creating a persistent request based on the search.

Amazon.com Delivers: Amazon.com Delivers is a variation on the Eyes feature. Customers select checkboxes to choose from a list of specific categories/genres (Oprah books, biographies, cooking). Periodically the editors at Amazon.com send their latest recommendations by email to subscribers in each category.

Bookstore Gift Ideas: The Gift Ideas feature allows customers to receive recommendations from editors. Customers pick a category of books for which they would like some suggestions. By navigating to that section of the "Gift Department," they can view a general list of recommendations created by the editors of Amazon.com. They also can select to view recommendations in one of a predefined list of categories including Globetrotter, Entrepreneur, and Teens. In many ways this serves as an online version of the Amazon.com Delivers feature discussed earlier. However, customers can be provided with recommendations anonymously since there is no need to register with the site as there is with Delivers.

Customer Comments: The Customer Comments feature allows customers to receive text recommendations based on the opinions of other customers. Located on the information page for each book is a list of 1-5 star ratings and written comments provided by customers who have read the book in question and submitted a review. Customers have the option of incorporating these recommendations into their purchase decision. Furthermore, customers can "rate the comments." With each comment is the question "Did this comment help you." Customers may indicate yes or no. Results are tabulated and reported such as "5 of 7 people found the following review helpful."

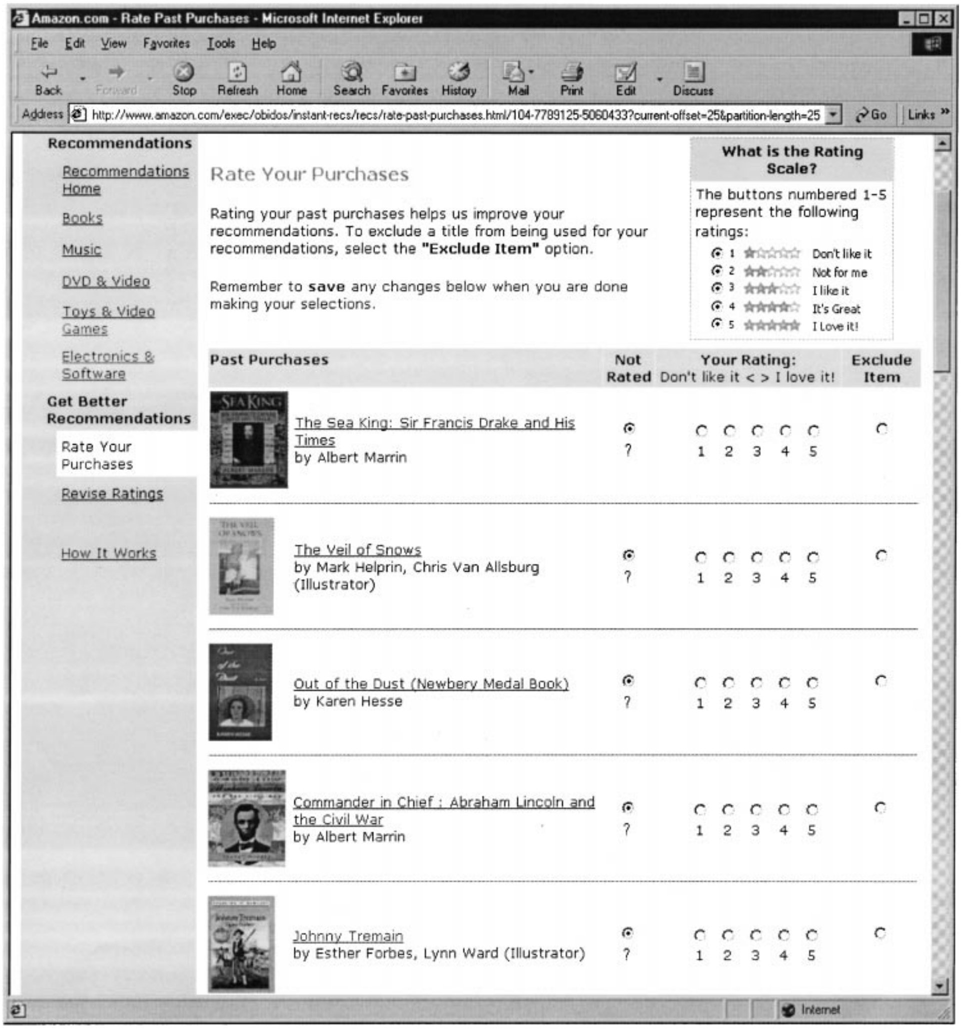

Figure 1. The Amazon.com ratings page prompts the customer to rate items recently purchased. These ratings are used as input to a recommendation engine to help the customer find other items that she is likely to like. Customers are asked to invest effort in rating, in exchange for which they get more useful recommendations.

Purchase Circles: The Purchase Circles feature allows customers to view the "top 10" list for a given geographic region, company, educational institution, government or other organization. For example, a customer could request to see what books are the best sellers for customers at Oracle, MIT, or residents of New York City. Purchase Circles provide another "fellow customer" form of recommendations by allowing customers not only to see what others are reading but also to personalize the recommendations by allowing them to select a “domain” with which they associate themselves. Customers can view Purchase Circles by navigating to the Circle that interests them.

3.2. CDNOW

Album Advisor: The Album Advisor feature of CDNOW™ (www.cdnow.com) works in three different modes. The first two are similar to the Customers Who Bought feature of Amazon.com. Customers locate the information page for a given album or artist. The system then recommends ten other albums related to the album or artist in question. Results are presented as “Customers who bought X also bought set S” or “Customers who bought items by Y also bought set T.” The third mode works as a “gift advisor.” Customers type in the names of up to three artists, and the system returns a list of ten albums CDNOW considers similar to the artists in question.

Related Artists: The Related Artists feature of CDNOW works on the assumption that if a customer likes a certain performer, there is a group of artists with similar styles that she will also like. Customers locate an artist and select the Related Artists link. Upon doing so, they are provided with a list of these artists who are considered to be “similar artists” and a list of artists who are considered to be among the “roots and influences” for the selected artist.

Buyer’s Guides: The Buyer’s Guide feature at CDNOW allows customers to receive recommendations based on a particular genre of music. Customers browse a list of genres provided by the site, including categories such as British Invasion, Big Chilling, and Parent Pop. Selecting one of the links from this list takes customers to a new list of albums the editors consider the essential part of this genre.

Artist Picks: In the Similar Artist feature at CDNOW one of the categories of recommendations is “roots and influences.” Presumably editors create this list. The Artist Picks feature provides similar recommendations, directly from the artists. Each week a different artist is featured, who lists the albums that shaped his or her taste as well as what is currently in their CD player.

Top 100: Traditionally, hype and “bestseller” status have been used by commerce sites to make recommendations to their customers. After all, if an album is on the Billboard Top 10, then it must be a good album. The Top 100 feature allows customers of CDNOW to receive this type of recommendation, but the 100 are drawn from the sales figures of the site and can theoretically be continuously upgraded to reflect actual sales.

My CDNOW: My CDNOW enables customers to set up their own music store, based on albums and artists they like. Customers indicate which albums they own, and which artists are their favorites. Purchases from CDNOW are entered automatically into the “own it” list. Although “own it” ratings are initially treated as an indication of positive likes, customers can go back and distinguish between “own it and like it” and “own it but dislike it.” When customers request recommendations, the system predicts six albums the customer might like based on what is already owned. Feedback is provided by customers selecting “own it,” “move to wish list” or “not for me” for any of the albums in this prediction list. The albums recommended change based on the feedback. Figure 2 shows a sample screen from My CDNOW.

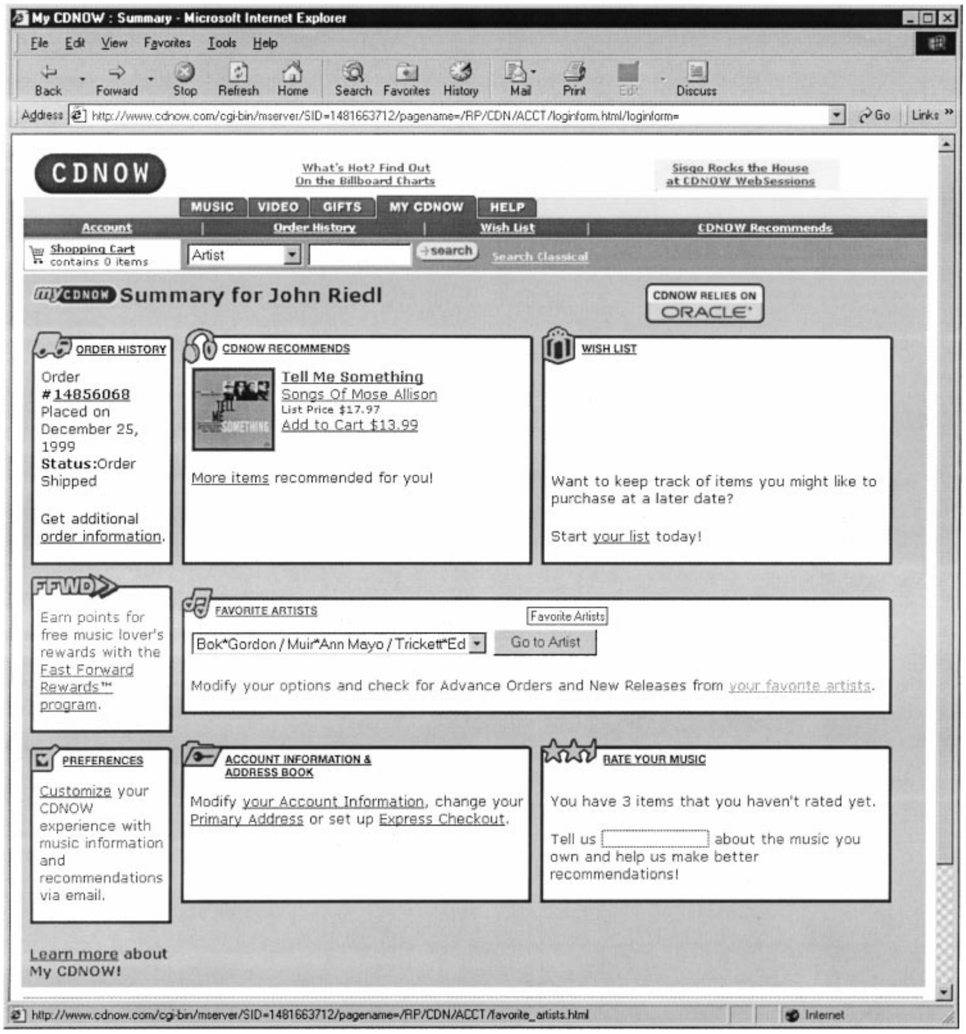

Figure 2. The My CDNOW page includes recommendations, prompts to rate past purchases, updates on favorite artists, and a Wish List, to communicate gift wishes to family or friends. This page serves as an entry point for a frequent CDNOW customer, as well as serving for an integrated access point to recommendations.

3.3. Drugstore.com

Advisor: The Advisor feature at Drugstore.com allows customers to indicate their preferences when purchasing a product from a category such as “suncare” or “cold and flu remedies.” For example, in the latter, customers indicate the symptoms they wish to relieve (runny nose and sneezing), the form in which they want the relief (caplets) and the “age” of patient to whom they want to administer the product (adult). Upon being provided with this information the Advisor returns a list of products recommended to meet the conditions.

Test Drives: In the Test Drives feature, a team of volunteers, made up of customers from the site, is sent a new product. These "fellow customers" provide reviews of the product including a star rating and text comments.

3.4. eBay

Feedback Profile: The Feedback Profile feature at eBay.com™ (www.ebay.com) allows both buyers and sellers to contribute to feedback profiles of other customers with whom they have done business. The feedback consists of a satisfaction rating (satisfied/neutral/dissatisfied) as well as a specific comment about the other customer. Feedback is used to provide a recommender system for purchasers, who are able to view the profile of sellers. This profile consists of a table of the number of each rating in the past 7 days, past month, and past 6 months, as well as an overall summary (e.g., 867 positives from 776 unique customers). Upon further request, customers can browse the individual ratings and comments for the sellers.

Personal Shopper: The personal shopper feature of eBay allows customers to indicate items they are interested in purchasing. Customers input a "short term" (30/60/90 days) and search on a set of keywords of their choosing, including their price limit. On a periodic basis (one or three day intervals) the site performs the customer's search over all auctions at the site and sends the customer an email with the results of this search.

3.5. MovieFinder.com

MovieFinder.com is the movie site maintained by E! Online.

Users Grade/Our Grade: Both the Users Grade and the Our Grade features report a letter grade recommendation to the customer. The Users Grade feature allows customers to register with the site and give letter grades (A-F) to the movies they have seen. These grades are then averaged over all customers and reported as the Users Grade. The Our Grade feature provides customers with a grade from the editors of E! Online. Thus, customers viewing the information page for Toy Story 2 might find that it gets a grade of A from the editors with a grade of A-from the customers who have rated it.

Top 10: The Top 10 feature at E! Online allows the customers to get recommendations from the editors in a category of their choice. Customers select a category from a list of previously defined categories such as chick flicks, sex scenes, and movies from books. Selecting a list takes the customer through descriptions of the top ten movies in that category as defined by one of the editors of E! Online.

3.6. Reel.com

Movie Matches: Similar to Amazon.com's Customers Who Bought, Reel.com's Movie Matches (www.reel.com) provides recommendations on the information page for each

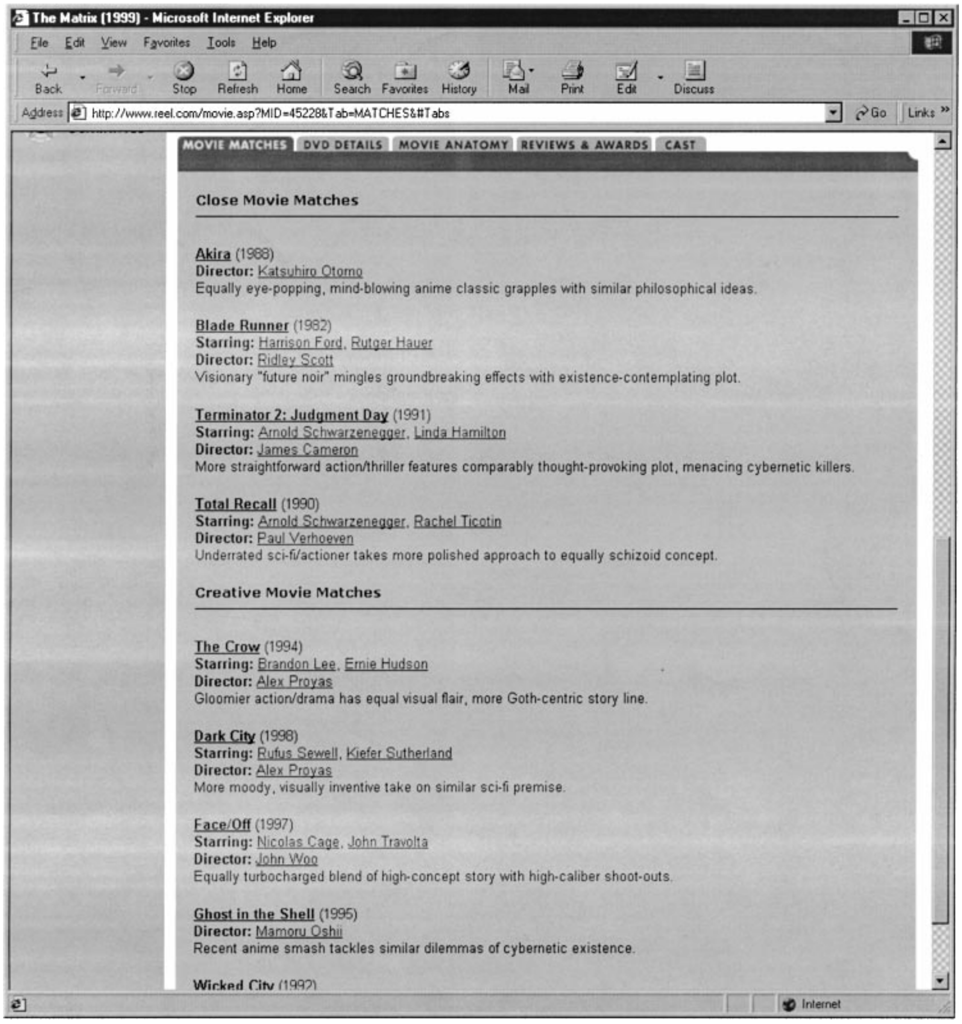

Figure 3. Reel.com's Movie Matches presents editors' picks for movies that will appeal to a customer using the movie a customer is currently browsing as an indication of interest. The picks are made by human editors, and come in two categories: Close Movie Matches, designed to provide safer recommendations, and Creative Movie Matches, designed to provide more serendipity in the recommendation.

movie. These recommendations consist of "close matches" and/or "creative matches." Each set contains up to a dozen hyperlinks to the information pages for each of these "matched" films. The hyperlinks are annotated with one-sentence descriptions of how the new movie is similar to the original movie in question (e.g. "Darker thriller raises similarly disturbing questions . . ."). Figure 3 shows a sample screen from Movie Matches.

4. A taxonomy for recommender applications

Using the previous examples, we have developed a taxonomy for E-commerce recommender applications that separates their attributes into three categories: the functional I/O, the recommendation method, and other design issues. Figure 4 illustrates how sub-parts of these categories are used in the recommendation process, as well as what the dimensions are within each portion. These parts and their dimensions are explained in more detail in the remainder of this section.

These three categories are not independent; as the examples in Table 1 illustrate, certain design choices require specific outputs. Similarly, certain outputs can be produced only by some, but not all, of the recommendation methods. Furthermore, our taxonomy includes only recommenders that attempt to help individual consumers based on preference data, such as interest information about products or purchase history. We do not consider methods that recommend according to less personal information, such as segmentation, demographic or psychographic or purely business information. Finally, we do not claim that the taxonomy is complete. Rather, it represents the range of E-commerce applications in use at the time of this writing. We fully expect new I/O, methods, and designs to emerge. We do, however, expect the basic structure of the taxonomy to remain useful as new practices are integrated into it.

4.1. Functional I/O

To simplify the process, we begin by concerning ourselves only with the data flowing into and out of these systems. Each system takes in a collection of inputs that may include consumer preference data, attribute data, and other correlates. Since this covers a large space of data, we additionally divide these inputs to indicate their origin—inputs about the targeted customer (i.e., about the customer for whom we are making recommendations) vs. general inputs regarding the community of other customers. Recommender applications use these inputs to produce output recommendations for other items. Analysis of these I/O produced the following dimensions.

Targeted customer inputs. Inputs about the targeted customer are fed into the recommendation process to provide personalized recommendations. An application that uses no inputs about the targeted customer can produce only non-personal recommendations. Adding one or more types of inputs allows the recommender application to personalize recommendations based on the customer's current activity, the customer's long-term preferences, or both. While there are multiple ways of categorizing the inputs from the targeted customer, one compelling set of categories evolves from the customer's approach toward providing the input.

While many recommender applications are still global in nature, more are beginning to respond to the customer's current state by using the customer's current navigation to provide context for the production or refinement of recommendations. Consumer behaviors interpreted for this input include both actions the consumer would have performed in exactly the same way even if he was unaware of the recommender system, and actions the consumer

Table 1. Taxonomy of recommender applications, organized by application model.

	Functional I/O				Design issues	
Application model	Targeted customer input	Community input	Output	Recommendation method	Degree of personalization	Delivery
Broad Recommendations List						
Amazon Bookstore Gift Ideas	Explicit navigation	Item attributes External item popularity	Suggestion	Manual selection	Non-personalized	Pulling unordered list and expert narrative
Amazon Purchase Circles	Explicit navigation	Purchase history	Suggestion	Statistical summarization	Non-personalized	Pulling ordered list
CDNOW Buyer's Guides	Explicit navigation	Item attributes External item popularity	Suggestion Reviews	Manual selection	Non-personalized	Pulling unordered list
CDNOW Artist Picks	None	None	Suggestion	Manual selection	Non-personalized	Pulling expert narrative
CDNOW Top 100	None	Purchase history	Suggestion	Statistical summarization	Non-personalized	Pulling ordered list
Drugstore Advisor	Keyword/Attributes	Item attributes	Suggestion Reviews	Attribute based Manual selection	Ephemeral	Pulling unordered list and expert narrative.
MovieFinder Top 10	Explicit navigation	Item attributes External item popularity	Suggestion	Manual selection	Non-personalized	Pulling unordered list
Comments and ratings						
Amazon Customer Comments	Implicit navigation	Ratings Text comments	Prediction Ratings Reviews	Statistical summarization	Non-personalized	Passive delivery of comments, individual ratings of other customers, and a predicted rating.
Drugstore Test Drives	Explicit navigation	Ratings Text comments	Prediction Ratings Reviews	Statistical summarization	Non-personalized	Pulling comments, individual ratings of other customers, and a predicted rating.

eBay Feedback Profile	Implicit navigation	Ratings Text comments	Prediction Ratings Reviews	Statistical summarization	Non-personalized	Passive delivery of comments, individual ratings of other customers, and a predicted rating.
Notification service						
Amazon Eyes	Keyword/Attribute	Item attributes	Suggestion	Attribute based	Persistent	Pushing single recommendation
Amazon.com Delivers	Keyword/Attribute	Item attributes	Reviews	Manual selection	Persistent	Pushing expert narrative
eBay Personal Shopper	Keyword/Attribute	Item attributes	Suggestion	Attribute based	Persistent	Pushing unordered list
Product-associated						
Amazon Customers who Bought	Implicit navigation	Purchase history	Suggestion	Item-to-Item correlation	Ephemeral	Passive delivery of unordered list
CDNOW Album Advisor-Single Item	Explicit navigation	Purchase history	Suggestion	Item-to-Item correlation	Ephemeral	Passive delivery of unordered list
CDNOW Album Advisor-Multiple Item	Keyword/Attribute	Purchase history	Suggestion	Item-to-Item correlation	Ephemeral	Pulling unordered list
CDNOW Related Artists	Explicit navigation	Item attributes	Suggestion	Manual selection	Ephemeral	Pulling unordered list
MovieFinder Users/Our Grade	Implicit navigation	Ratings	Prediction	Statistical summarization	Ephemeral	Passive delivery of predicted rating
Reel.com Movie Matches	Implicit navigation	Item attributes	Suggestion	Item-to-Item correlation	Ephemeral	Passive delivery of unordered list
Deep personalization						
Amazon Your Recommendations	Purchase history Ratings	Ratings Purchase history	Prediction Suggestion	User-to-User correlation	Persistent	Pulling (un)ordered list
My CDNOW	Purchase history Ratings	Purchase history	Suggestion	User-to-User correlation	Persistent	Passive delivery of unordered list

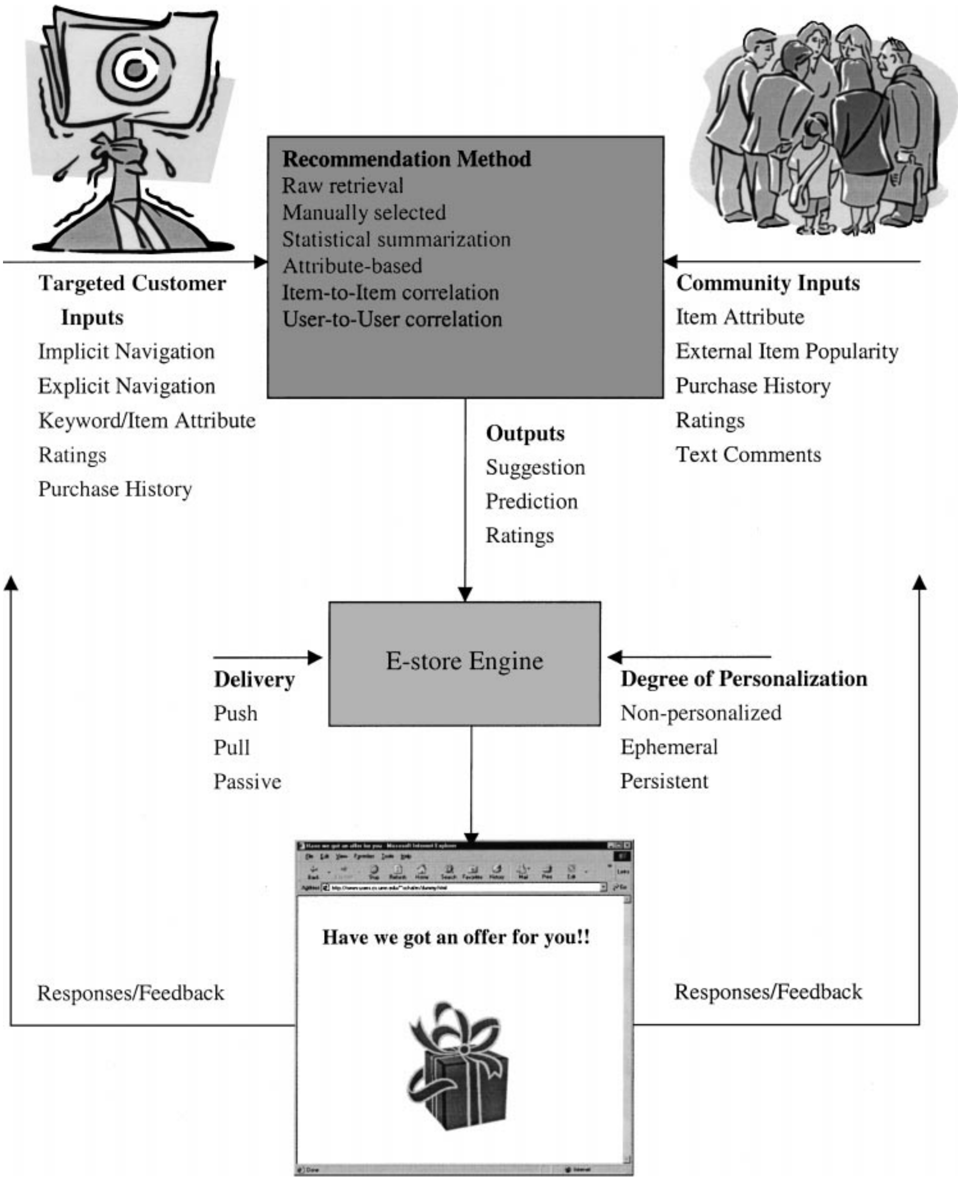

Figure 4. Recommender applications combine inputs about the customer in question, with those about product and user communities to generate recommendations. Sites use decisions about personalization level and delivery method to transform these into specific recommendation packages. Feedback to these recommendations may generate additional inputs for future recommendations.

performs for the sole purpose of enhancing the recommendations. *Implicit navigation* inputs are, generally, inferred from the customer's behavior without the customer's awareness of their use for recommendation processes. This input may include the *specific item* or items that the customer is currently viewing or those items in the customer's shopping cart. For example, Amazon.com utilizes the particular book that a customer is browsing to recommend a set of additional books considered in some way similar to the currently viewed text. This input may also include the *category or feature* to which the customer has navigated. In doing so, e-merchants hope these applications will help convince the browser that the initial product is worthwhile—if he likes the "similar" item(s)—and help sell multiple products at once.

In contrast, *explicit navigation* inputs are intentionally made by the customer with the purpose of informing the recommender application of his or her preferences. To offer these, sites provide the customer with a finite set of attribute choices as navigational links. For example, a customer using MovieFinder's Top 10 feature is provided with a hyperlinked list of top ten lists produced by the editors. By navigating to a list of interest, the customer can get recommendations for products in a fairly specific category. Despite differences in the configuration of these systems, from a customer's point of view, he is simply navigating.

In some cases, input from the customer can not be limited to a single category or item of interest. In these cases, applications may use *keywords and item attributes*, either explicitly from a search or implicitly, derived from the items being viewed. In either case, these keywords/attributes are interpreted as input that models the customer's current interests. For example, customers using the Advisor at Drugstore.com provide information about their wants and/or needs before receiving recommendations for products such as cold and flu remedies. Systems using these types of inputs replace the feel of navigation with the feel of searching.

The targeted customer may provide the most helpful and explicit inputs in the form of ratings of items she has consumed. In an ideal situation, customers are presented with a representative sample of items from the e-merchant's database and are asked to indicate their preference for each of the representative items. This can consist of numerical ratings ("rate each on a scale of 1–5"), or a simple binary rating ("did you like this?"). Customers who create a personalized My CDNOW are given the opportunity to indicate explicitly the albums that they already own, separating them into the ones they like and the ones they wish they had never purchased. In doing so the customer goes through a task that feels like neither navigating nor searching. Rather, the feel is that of configuring. The customer is providing data to the site to allow the business to provide a more personalized experience.

Rather than asking consumers to provide explicit ratings, some sites utilize the targeted customer's *purchase history* as an implicit form of ratings. These provide lists of items for which the customer has expressed a very concrete preference. For example, once a customer sets up her My CDNOW account, all additional purchases are recorded in the "bought it and liked it category." This input, however, has no real "feel" to the customer. She is simply using the site. Good implementations of purchase histories recognize that they are related to ratings and allow the customer to enter them in "ratings mode." For example, users of My CDNOW can review their ratings (including those entered implicitly through purchase) and change the "liked it" to "hated it."

Community inputs. Community inputs include a broad range of data regarding how multiple individuals in the community, or the community as a whole, perceive items. Inputs that reflect overall community opinions include *item attribute* assignments that assign community-based labels and categories to items. For example, many attributes such as film genre and book categories reflect the consensus of the broader society. Similarly, *external item popularity* may reflect popularity in broader communities such as global box-office sales or national best-seller lists. In manually selected recommendation lists such as CDNOW's Buyer's Guides, it is presumed that editors are taking into account more than site sales figures to generate their list of most popular products. Finally, just as we described using the purchase history of an individual customer as a set of implicit ratings about products, we can utilize the *community purchase history* to do the same. These can be combined to produce site-specific top seller lists (Purchase Circles), or mined to discover similarities, and draw conclusions about sales trends or item similarity (Album Advisor).

While the previous community inputs are tied to the community as a whole, other inputs are directly associated with individual members of the recommender community. Several sites encourage *text comments* from their customers. Systems such as Drugstore.com's Test Drives gather comments about a single product and present these as a means to facilitate the decision-making process. While text comments are helpful, they require a fair amount of processing by the targeted customer. The customer must read each paragraph and interpret to what degree it is positive or negative. To simplify this process, most sites offering the opportunity for the community to write text comments also encourage the members to indicate some form of numerical score or *ratings*. Just as recommendation systems can use the ratings of the targeted customer, they can also gather the ratings of all customers to provide data for use in producing recommendations.

Worth discussing at this time is the source of these inputs. Most of the sites in our survey appear to be using largely site-specific data about their customers, combined with both site-specific and syndicated data about their products. Item attributes are usually syndicated through services that publish digital catalogs comprising categorizations and descriptions of products. For instance, book vendors often use third party genre and keyword classifications. These third party attributes are often supplemented with a smaller amount of site-specific data. External item popularity is nearly always syndicated to provide a broad measure of consumer interest. Community purchase information is always site-specific, based on purchase behaviors of groups of customers from the site. Community text comments and ratings are primarily site-specific. In principle these data could be shared between sites, but we do not know of examples of such sharing to date.

When syndicated data about products are obtained, it must be unified with data from the site, such as the site's catalog and editor's assessments. Syndicated data about individual consumers can also be purchased and used; this is common with mailing lists and demographic data. In this case, the unification is even more challenging than when using product data, because the providers of the data—the consumers—are often resisting the unification. (For instance, consumers may provide false information to protect their privacy.) Furthermore, consumers may resist such unification in the press or courts as in a recent case in the online advertising industry. Unification of syndicated data clearly has the potential to enhance recommendations for consumers; whether or not it will be common in practice is still an open issue.

Outputs. Output recommendations of specific items vary in type, quantity, and look of the information provided to the customer. The most common type of output can be considered a *suggestion*. This often takes the form of "try this," or simply placing "this" in the web page viewed by the user. The simplest form of "this" is the recommendation of a single item. By recommending only a single item, the e-merchant increases the chance that the customer will seriously consider the item since the recommendation takes little time to process. However, it also places all of the risk in a single recommendation, which may be rejected because the customer already owns the item or has other outside knowledge. Targeted advertising also generally results in an individual recommendation as do "check-out coupon" promotions and some other systems designed to elicit up-selling. More commonly, recommender systems provide a set of suggestions for a customer in a particular context. Some application designers prefer to leave the list unordered, to avoid giving the impression that a particular recommendation is the best one. Unordered lists may avoid premature customer dismissal of an entire set of recommendations based on rejection of the first one. (Of course, every list has some order; many "unordered" lists are deliberately presented in another order, such as alphabetical order, to avoid being misinterpreted as best-first.) Other applications rank recommended items. The ordered list provides extra information that may be helpful to customers.

Several recommender algorithms present consumers with a *prediction* of the rating they would give to an item. These estimates can be presented as personalized estimates for individual consumers or as non-personalized estimates for typical community members. These predicted ratings can help customers understand the strength of a recommendation. Predicted ratings can be displayed in the context of individual recommendations or lists of recommendations, or they can be displayed in the context of general item information. MovieFinder's "Customer Grade/Our Grade" feature provides two different predictions (community and editorial) on an A to F scale that are presented as a user browses the information screen for a movie of her selection.

When communities are small or community members are well known, it may be useful to display the individual *ratings* of community members to allow the targeted customer to draw her own conclusion about the strength of a recommendation. This technique is particularly valuable when the customer can select known community members or when the ratings are accompanied by *reviews*. Reviews are an example of recommendations that contain evaluations that are not completely machine-understandable. Indeed, unlike other recommendation techniques, it is difficult to distinguish text comments that recommend for and against a particular item, though, as previously mentioned, many systems that use text comments also ask reviewers to include a numerical rating. Presenting text comments to customers provides them with an understanding of why a particular item should be favored or disfavored, and comments may be the only way to help a customer navigate through substantial disagreement among people who have agreed before. Amazon.com and eBay both help people evaluate items (books and commerce partners) by presenting text comments and ratings in a non-personalized way. That is, each customer sees the same, complete set of comments. It is possible to select or order comments based on a customer's history of agreement with the commenter, but we are unaware of any E-commerce applications that do so. By doing so, these systems would be providing commenters with "credentials"—some indication that this person's comments hold value. The closest step we've seen is a level of "meta-rating" on Amazon.com where readers of comments can rate the comments themselves. There is as yet no mechanism for automatically using those ratings to create personalized sets of comments.

4.2. Recommendation method

In the previous section we focused on the data utilized and generated in the recommendation process. In this section we provide an overview of the specific processes used in actual E-commerce recommender systems. We should point out that individual systems may actually use a combination of these processes. Each category discussed here represents a family of algorithms and approaches. It is beyond the scope of this work to survey the variant implementations of each process. Breese et al. (1998) compare a variety of algorithms for recommendation generation, and Herlocker et al. (1999) provide a detailed comparison of customer-to-customer correlation algorithms.

The *raw retrieval* "null recommender" system provides customers with a search interface through which they can query a database of items. In this case, recommendation is a "binary," syntactic process whereby the system "recommends" whatever the customer has requested. While not technically a recommender application, such an application may appear as one to customers. For example, when a customer asks a music site for albums by "The Beatles," the system returns a list of Beatles albums that may be helpful and may indeed lead the customer to an album of which he was previously unaware. Raw retrieval systems are ubiquitous in E-commerce applications.

Applications that value personality over personalization may create sets of recommendations that have been *manually selected* by editors, artists, critics, and other experts. These "human recommenders" identify items based on their own tastes, interests, and objectives and create a list of recommended items available to community members. Their recommendations are often accompanied by text comments that help other customers evaluate and understand the recommendation. For example, customers using the MovieFinder Top 10 lists select a particular "genre" for which they would like recommendations—for example, "chick flicks." They are provided with a list manually compiled by an editor listing what she considers to be the top ten chick flicks of all time. The process does not use computer computation at all, but simply reproduces what could appear in a list on the wall of any video store. This process most closely mimics traditional critics and editors, including both potential insight and potential bias. Although not included in our examples, an increasing number of sites allow any community member to establish recommendation lists (Art.com).

In cases where personalization is impractical or unnecessary, recommender applications can very efficiently provide *statistical summaries* of community opinion. These summaries include within-community popularity measures (e.g., percentage of people who like or purchase an item) and aggregate or summary ratings (e.g., number of people who recommend an item, average rating for an item). They include systems such as eBay's customer feedback, which provides average ratings of buyers and sellers. Prospective sellers and buyers can consult the average and the individual evaluations but cannot see the rating by "customers I've agreed with." While these summaries provide only non-personalized recommendations, they are popular because they are easy to compute, and they can be used in non-customized environments such as physical store displays.

Recommendations based on the syntactic properties of the items and customer interests in those properties use *attribute-based* recommendation technologies. Though the simplest attribute-based recommendation is raw retrieval, true "recommenders" that use attributes model customer interests beyond a simple query. For example, a customer who is browsing in the "country music" section of a music store and who has several "$9.99 special" compact disks in her shopping cart might receive recommendations for discount country CDs. Other attribute-based recommenders use customer profiles that indicate likes or dislikes to make recommendations to the customer. For example, the same music store may learn that a particular customer only buys discounted CDs or that another customer never buys music from the 1970s.

Other applications use *item-to-item correlation* to identify items frequently found in "association" with items in which a customer has expressed interest. Association may be based on co-purchase data, preference by common customers, or other measures. In its simplest implementation, item-to-item correlation can be used to identify "matching items" for a single item, such as other clothing items that are commonly purchased with a pair of pants. More powerful systems match an entire set of items, such as those in a customer's shopping cart, to identify appropriate items to recommend. Item-to-item correlation recommender applications usually use current purchases or other current interests rather than long-term customer history, which makes them particularly well-suited for recommending gifts. A customer merely needs to identify some other items liked by the recipient to elicit gift recommendations tailored to the recipient rather than the giver.

Finally, recommender systems using *user-to-user correlation* recommend products to a customer based on the correlation between that customer and other customers who have purchased products from the E-commerce site. This technology is often called "collaborative filtering" because it originated as an information filtering technique that used group opinions to recommend information items to individuals (Resnick et al., 1994; Hill et al., 1995; Shardanand and Maes, 1995; Konstan et al., 1997). My CDNOW is a system that uses user-to-user correlations to identify a community of customers who tend to own and like the same sets of CDs. The principle is that if several members of my community owned and liked the latest Sting album, then it is highly likely that I will too. Though we use the word correlation in the name of this technique, thus hinting at nearest-neighbor techniques based on linear correlation, the technique can be implemented with many other technologies as well (Breese et al., 1998).

One important issue when considering the recommendation method is whether the computation can be performed entirely online, while the Web store is interacting with the customer, or whether parts of the computation must be performed offline for performance reasons. Online recommendations are preferred because they can respond immediately to the consumer's preferences. Most of the recommender processes mentioned above can be performed entirely online. Raw retrieval, manual selection, statistical summarization, and attribute-based are all simple computations that are usually performed during customer interaction. Item-to-item correlation and user-to-user correlation are computationally more intensive and often require an offline component to prepare a model that can be executed efficiently online. One challenge in designing the model-building is to ensure the resulting online system is as responsive as possible to interactive input from the user.

4.3. Other design issues

Degree of personalization. Recommender applications may produce recommendations at varying degrees of personalization. The degree of personalization encompasses several factors including both the accuracy and the usefulness of recommendations. Accuracy measures how correct the system is while usefulness includes such factors as *serendipity*—whether the system provides valuable, but unexpected recommendations-and *individualization*—whether the system provides different recommendations to different people—measures which are both important. An accurate system that only recommends consensus best-sellers provides less value than a system that can find and recommend more obscure books of interest to particular users. Similarly, a system that recommends obscure books, but that is rarely correct, would not be used for long. While personalization is a continuum across several dimensions, we find it useful to specifically identify three common levels.

When recommender applications provide identical recommendations to each customer, the application is classified as *non-personalized*. The specific recommendations may be based on manual selection, statistical summarization, or other techniques. Many of the E-commerce recommendation examples are non-personalized. Top-sellers, editor choices, average ratings, and unfiltered customer comments all present the same recommendations to each customer of the system.

Recommenders that use current customer inputs to customize the recommendation to the customer's current interests provide *ephemeral personalization*. This is a step above non-personalized recommenders because it provides recommendations that are responsive to the customer's navigation and selection. Particular implementations may be more or less personal, however. A recommender application with a high degree of ephemeral personalization would be one that uses an entire current browsing session or shopping cart to recommend items. Conversely, a recommender application that simply attaches recommendations to the current item is nearly non-personalized. Ephemeral personalization is usually based on item-to-item correlation, attribute-based recommendation, or both. Examples of ephemeral personalization include CDNOW's multi-item Album Advisor and certain versions of Drugstore.com's Advisors. Both take information provided by the customer at recommendation time and return a list of suggestions from that ephemeral context.

The most highly-personalized recommender applications use *persistent personalization* to create recommendations that differ for different customers, even when they are looking at the same items. These persistent recommenders employ user-to-user correlation, attribute-based recommendation using persistent attribute preferences, or item-to-item correlation based on persistent item preferences. They require customers to maintain persistent identities, but reward them with the greatest level of personal recommendation. Examples of persistent personalization include My CDNOW, which uses user-to-user correlation, and

Amazon.com's Eyes and eBay's Personal Shopper, which use persistent attribute recommendation.

Delivery. Matching the delivery of recommendations to the customer's activity is a critical design decision in E-commerce recommender systems, just as it is in traditional marketing. In fact, E-commerce provides close analogues to traditional solicitation and retail models. Marketers have long used direct mail and outbound telemarketing in an attempt to get customers to initiate a new buying session. *Push* technologies have the benefit of reaching out to a customer when the customer is not currently interacting with the e-merchant. In E-commerce applications, e-mail is the most commonly used push technology for delivering recommendations. Sending recommendations, and perhaps promotional offers, invites the customer to return to the e-merchant. Indeed, today's technology allows customers to click on a link in the e-mail message and be taken directly to the recommended product on-line.

Applications using *pull* technologies allow the customer to control when recommendations are displayed. They make the customer aware that recommendations are available (e.g., by displaying a link to them) but do not actively display recommendations until the customer requests them. This request may appear in different contexts, such as a request to evaluate a specific product, a request to find a gift, or a request for recommendations in a category. Some early applications used pull delivery because recommendation computation was expensive and could slow down the interactivity of a site. Today, pull delivery is a design choice for applications where types of recommendations are considered peripheral (e.g., top 10 lists or gift recommendations) rather than integrated into the application.

Sometimes referred to as "organic" recommendations, *passive* delivery presents the recommendation in the natural context of the rest of the E-commerce application. Examples of passive recommendation include displaying recommendations for products related to the current product (Amazon.com's Customers Who Bought feature), displaying recommendations for products related to the topic of a text article (CDNOW's Artist Picks) and displaying recommendations in the context of exploration (Drugstore.com's Advisors). Passive recommendation has the advantage of reaching the customer at the time when the customer is already receptive to the idea. Indeed, E-commerce uses passive recommendation as part of the ordering process, suggesting upgraded shipping options, for example, at the time when the customer is completing a purchase (where it is much more effective than asking about shipping on a link off the home page). A possible disadvantage of passive recommendations is that customers may not actively notice them, but we are not aware of any research that suggests that noticing recommendations explicitly makes them more effective than having them as part of the overall experience.

We find that the preferred methods of delivery are changing. Early applications focused on push and pull delivery because of performance and the desire to show customers that "they care" and are actively recommending. More recently, applications have been shifting to passive and push delivery—passive delivery on their web site with pushed recommendations to bring customers back.

5. E-commerce recommender application models

Section 4 classifies e-commerce recommendation applications by the functional inputs and outputs to the application, the recommendation method, and other design issues. Table 1 reveals five patterns in e-commerce recommender application designs, each addressing different business goals. This section identifies the five business goals and the application models used to address them. Designers of e-commerce recommender applications can use these models as examples of already-proven solutions to be emulated or can use these as a base from which to explore as-yet-untested recommender application models that may address new business needs.

Helping new and infrequent visitors: Broad recommendation lists

One of the key challenges for e-commerce sites is to engage visitors—especially new and infrequent visitors—before they leave to visit another site. For sites that list thousands to millions of different products, this challenge is particularly acute; they must not only engage the visitor but also keep him from getting lost and frustrated. Nearly every site visited has some form of broad recommendation list designed to direct customers towards engaging products. These lists typically allow the targeted customer to use current navigation to pull non-personalized suggestions. These include overall best sellers, best sellers in a category, editor and expert recommendations, and other collections of products selected either manually or through simple statistical summarization. In essence, these recommendation lists replace the tabletop displays, endcaps, and large product displays in physical stores. Whichever technique is used, the lists help orient users who might otherwise leave before finding compelling products.

Part of what makes broad recommendation lists so prominent is the low level of needed input; no personal information is needed, except for minimal ephemeral context about the category of interest to the customer. Broad recommendation lists allow marketers to adjust pricing and inventory to match the recommendations since they can be assured that these recommendations will reach a large audience. Editors or experts can create text to surround broad recommendations to market the recommended products to customers. The recommendations themselves can be delivered in several different ways, though most applications either place them on a home page or “category home” page or advertise them and have users directly select them.

Building credibility through community: Customer comments and ratings

Retailers in general, and e-retailers in particular, must often overcome an image of low credibility. Customers may feel that the site is interested only in making a sale, and therefore that it will present any “recommendation” or advertising necessary to induce them to make a purchase. While principles of one-to-one marketing suggest that it is in the retailer’s interest to serve the interests of the customer, stores must still leap over the credibility hurdle to move towards a one-to-one relationship. One way to do this is to collect reviews and ratings from members of the community at large. These systems use the targeted customer’s current navigation to suggest which non-personalized reviews, ratings, and predictions to display passively. By building a "community center," sites allow customers to communicate with each other and provide each other with advice and feedback on products.

These "grass roots" recommendations require little site-directed effort, since the customers do all of the evaluation. They also provide a high degree of credibility since customers often are more likely to believe a set of other customers than the marketer who makes money on the purchases. As a side benefit, they create a sense of community that can distinguish the site from others and thereby retain customers. Customer comment applications provide a summary of the ratings, either as an average or another figure representing the rate of positive and negative recommendations, and give the customer an opportunity to read through the ratings and form her own opinion.

Inviting customers back: Notification services

Stores that know their customers' interests can leverage that information by inviting them back to the store when new products of interest arrive or are discounted. Notification services use keywords provided by the targeted customer and attributes of the items being recommended to push persistent, personalized suggestions and can thereby build stronger customer relationships. Many e-merchants allow customers to describe the products they find interesting and then automatically notify them when such products are available. These notification services can provide a great service to the customer, who becomes quickly aware of new products of interest, and can be very effective at bringing customers back to the e-commerce site on a regular basis. The form of the descriptions can vary from a simple keyword or attribute query to a more complex specification that includes price ranges.

Cross-selling: Product-associated recommendations

Suggestive selling is particularly effective when the seller knows the current interests of the buyer. Retailers arrange products to enhance cross-selling by placing complementary items in close proximity. On-line retailers are freed from physical layout and can directly suggest products related to the one a customer is viewing. By using the targeted customer's current navigation as an ephemeral indication of interest, such systems use item-to-item correlation and community purchase history to passively display suggestions to the targeted customer.

Many different recommender applications use the context of a current product or several current products to recommend other products using a variety of recommender methods. This popularity is partly due to the variety of inputs that can be used to generate such recommendations, including anonymous purchase histories, customer purchase histories, ratings, product attributes and expert opinions. Product-associated recommendations are particularly well suited for passive delivery since they can be integrated into a product information page.

Building long-term relationships: Deep personalization

The goal of most retail businesses is to develop long-term relationships with customers that lead to higher lifetime values and greater competitive barriers. Deep personalization, based on a customer's history of preferences, purchases, or navigation, is the strongest and most difficult type of personalization to implement. Deep personalization is common already in web advertising and is becoming more widely used in e-commerce now that collaborative filtering recommendation engines are readily available. By utilizing collaborative filtering's ability to match the targeted customer's history with histories of other customers, deep personalization is able to generate persistent, personalized suggestions or predictions. Deep personalization builds a customer relationship over time, leveraging the history developed to provide increasingly better recommendations. Unlike notification services that require manual updating, deep personalization updates the user profile whenever the customer interacts with the merchant. Deep personalization systems can use user-to-user correlation, attribute-based systems with a learning module to identify user interests, or a combination of the two.

6. Research challenges

Although commercial recommender systems have been available for several years now, challenging research problems remain. This section outlines some of the more important technical challenges and when appropriate, describes recommender system applications that face the challenge as well as the opportunity that can be realized by overcoming the challenge.

Scalability and real-time performance

Scalability in recommender systems includes both very large problem sizes and real-time latency requirements. For instance, a recommender system connected to a large Web site must produce each recommendation within a few tens of milliseconds while serving hundreds or thousands of consumers simultaneously. The key performance measures are the maximum accepted latency for a recommendation (tens to hundreds of milliseconds), the number of simultaneous recommendation requests (tens to thousands), the number of consumers (hundreds of thousands to millions), the number of products (tens to millions), and the number of ratings per consumer (tens to thousands). Many techniques from data mining can be adapted to the scalability problem for recommender systems, including dimensionality reduction and parallelism, but must be modified to meet the simultaneous throughput and latency requirements.

A further challenge in adapting scalability techniques from data mining is the extreme sparsity of ratings in recommender systems. The value of a recommender system lies in the fact that most customers have not evaluated most of the products. A typical bookstore customer may rate thirty books and seek recommendations from over a million books in print. While thirty books may represent a substantial opinion, will there be enough other customers whose reading histories overlap with this .003% of the catalog? Researchers are exploring a variety of techniques to bridge the gaps caused by sparsity. Examples include explicitly supporting transitivity in neighborhood formation and using dimensionality reduction to shrink the effective dimensionality of the product space. Many classical dimensionality reduction techniques work poorly with extremely sparse data, so these techniques will have to be modified to work for recommender systems.

Ironically, not enough data is also a challenge for recommender systems, which must collect enough data to make effective recommendations for new customers. Sites can make good recommendations faster if they share information about their customers with other sites. Shared information benefits customers who receive more accurate recommendations in less time. Sites that use shared information benefit because the better recommendations help their customers find more products to purchase and because the customers prefer to return to stores with the better recommendations. Large sites that have a wide variety of customer data tend to prefer not to share the data with competitors because they feel having the data gives them a competitive advantage. Smaller sites can band together to overcome this competitive disadvantage by sharing customer data. Further, a consortium of non-competing sites may form with the goal of sharing data to increase the value to companies within the consortium. Customers of sites that share data will need assurances that their privacy will be carefully protected, even as their data are shared beyond the boundary of a single site. The appendix discusses privacy issues in recommender systems.

Incorporating rich data

Recommender applications currently use a wide range of input data in forming their recommendations, including explicit ratings and simple behavioral data such as purchases and click-throughs. However, there are many other types of data that can be collected and used. In the future, recommender systems will commonly collect dozens of different types of data and integrate them into effective recommendations. Ongoing research in web usage mining and more general commerce-related data mining may reveal techniques for exploiting complex behavioral data.

Some recommenders, such as those on news sites, will want to provide recommenders that incorporate content analysis as well as preference analysis. Many machine-learning techniques have been shown to effectively build a model of human preferences based on content. Information filtering systems (Salton, 1968) build models of customer preferences based on keyword vectors. Other systems employ neural networks and other feedback systems to learn preference patterns. While these systems cannot entirely replace the human values captured in collaborative filtering systems, there is evidence that they can augment collaborative filtering. Other approaches combine machine-learning techniques with collaborative filtering ones. For example, Basu (1998) shows how Ripper, a rule induction system, can be used to create classifiers from ratings and content data to make personal recommendations. Fab (Balabonovic, 1997) uses a two-level architecture with content-seeking agents that are trained by a variety of techniques and personalized user agents that select streams of content for each user. Our research has shown that the inclusion of filtering agents can increase both the accuracy of recommendations and the percentage of items in the database for which recommendations can be generated (Sarwar et al., 1998; Good et al., 1999). Thus far, research has focused on the use of “generic” syntactic agents (e.g., an agent that likes horror films) and personal agents trained for the target customer using rule induction and information retrieval (TFIDF) techniques. New work is underway to explore the value to customers of having access to agents trained for other customers (i.e., those that try to mimic others but do so by building a preference model based on content features and therefore can rate all items). Adding other machine learning systems to recommender systems may increase the accuracy of the recommender system, especially in cases where few humans have tried the products in question.

While many products appeal to consumers over a long time period, others have temporal relevance. Seasonal goods such as snow shovels and lawn care products may be highly desirable one month and useless two months later. Products for children exhibit similar patterns; the parent who buys toys for an infant should not receive infant toy recommendations for the same child two years later. Data mining research has addressed some issues in sequence identification and temporal associations. How can these results be used to influence recommendations to make them more helpful to consumers?

Consumer-centered recommendations

One important difference between a recommender system and a traditional data mining system is that the end-user for the recommender system is the consumer. This difference leads to several desirable properties for recommender system algorithms that have not been explored in data mining algorithms.

Some recommendations are most valuable when they apply to a group of consumers rather than an individual. For instance, movies are most often attended socially. The choice of the right "date movie" can be very important. Recommenders can be used to select products that maximize the value of the product to a group of people. Some systems already support simple versions of this idea, such as selecting a movie that two people will like. Future multi-user recommendation systems will let customers control how the recommender system balances their interests in choosing a product. Should it choose a movie that neither will hate? That she will love? That they will both like? Different algorithms are needed for each of these scenarios to maximize the social good for the group, according to their needs.

The success of a recommender system should be measured by how effectively the system helps customers make decisions that they, retrospectively, consider correct. One interface innovation that can help customers decide when to follow recommendations is to explain the recommendation to the customer, just as many machine-learning algorithms explain their results to their users. Researchers are currently experimenting with several different explanation models, including summarizing the data behind recommendations, explaining recommendations indirectly (e.g., indicating which of the customer's own preferences most strongly led to a particular recommendation), and providing "persuasive" evidence about the system's success in similar recommendations. Simpler explanation systems display a brief capsule of the amount of data or expected variance in a prediction. Additional research is needed regarding explanation algorithms for other recommender algorithms, and on the effectiveness of explanations in helping customers decide among recommendations and increasing customer confidence in using recommender systems.

Connecting recommenders to marketers

Recommender systems are currently used as virtual salespeople, rather than as marketing tools. The recommender system makes good decisions locally, but it is neither able to give the type of feedback needed for marketing professionals, nor is it able to accept input from marketing professionals about global preferences. In the future, recommender systems must integrate marketing features more completely.

Systems based on association rules or other non-personal data mining approaches build models of customer preference and must then filter out products the customer already owns to avoid seemingly trivial recommendations. Personalized recommender systems include customer ratings and purchases in the process and thereby avoid this challenge, but they too could benefit from more effective filtering of recommendations. Frequently either the marketer or the customer wants a recommendation set that is restricted based on specific criteria. For instance, a consumer might want recommendations only for products that are currently in stock, or a marketer might want to avoid recommending an R-rated movie to an eleven year old. Today's recommender applications must post-filter recommendations or pre-filter candidates for recommendation. Research is needed to better integrate the recommendation process with other forms of product filtering to allow more integrated solutions.

Recommender systems should also be integrated with the marketer's reporting systems. Recommender systems target each individual customer differently, making it difficult to produce the reports to which marketing professionals are accustomed. These reports usually partition the population into a manageable number of segments. One way to bring these two worlds together would be to use the people-to-people correlations implemented by some recommender system algorithms to create segments for the reports. Open questions include "How can names be assigned to the automatically generated segments?" and "Are automatically generated segments more useful for managing marketing campaigns than traditional segments?"

Recommender systems can be made more useful as marketing systems in other ways as well. Current recommender systems are mainly "buy-side" systems; that is, they are designed to work on behalf of the consumer to help him decide what products he should purchase. However, modern marketing is designed not only to maximize utility to the customer but also to maximize value to the business at the same time. The recommender system could produce an indication of the price sensitivity of the customer for a given product. For instance, one customer might be willing to purchase the product at a price that would earn the site ten cents of profit while another customer might purchase the same product at a one-dollar profit. Furthermore, this price sensitivity indication could be combined with a lifetime value model (e.g., Mani, 1999) to determine when it is in the best interest of the site to sell a product at a lower price-potentially even below cost-to increase the customer's loyalty and thereby her long term value to the site. There are challenging ethical and practical issues in implementing systems like these that use information from studying the customer to implement price discrimination. The ethics of opinion ownership may lead sites to directly compensate customers for their information (Avery, 1999).

How will recommender systems that are serving marketers evolve to protect the trustworthiness that makes them valuable to customers? Recommender systems emerged as agents to help customers find products to purchase. These recommenders were implemented by sites to create additional value for their customers, so those customers would be more loyal to the site. Now, some recommenders are evolving towards marketing systems that help sites sell high profit products. One way to counter this evolution is to create verifiably unbiased recommender systems that consumers trust to recommend on their behalf. In the mid 1990s Firefly proposed creating such a branded recommender system. Consumers may be suspicious that an unscrupulous Web site might modify the outputs of such a recommender before displaying them. How can a trusted recommender validate itself to consumers through a Web client? Is it possible to build portable recommenders that can run on consumer's palmtop computing devices, serving only the consumer?

For small collections of items, such as movies, it is possible to produce recommendations for a large set of items in advance and simply display them on-site. In theory, an off-line recommender could store the ratings profiles from a neighborhood of customers, but in practice the level of ratings sparsity makes this approach infeasible. Is it possible to build an entire off-line recommender system that can accept new ratings from the target customer and generate real-time recommendations? Such a recommender might include the use of filtering agents trained to replicate the rating pattern and the generation of compressed ratings datasets through dimensionality reduction techniques. Consumers would know they could trust their own live recommender system loaded with a highly compact representation of the opinions of a community of trusted consumers.

7. Conclusion

Recommender applications address a variety of E-commerce business needs. They allow businesses to practice mass customization (Pine, 1993) by creating a customized experience through a set of standard products and by allowing product components to be assembled into customized products. As businesses focus on long-term customer value (Peppers and Rogers, 1997), they need advantages that help them retain customers. In E-commerce, the advantage of location is gone, and businesses must depend more heavily on information advantages. Recommender systems allow businesses to leverage their customer history to create more personalized experiences for their customers. Those customers will quickly discover that the business that "knows them best" is the one that can serve them most effectively, recommending the right products rather than treating them like strangers.

In this paper, we have surveyed the recommender applications used by several of the largest E-commerce companies. We identified several design parameters and developed a taxonomy that classifies these applications by their inputs, output, recommendation method, degree of personalization, and delivery method. Classifying the applications revealed a set of application models that reflect the state of practice. We have also explored promising directions in recommender systems, including application ideas built on innovative models that transcend current practice. Finally, in the appendix, we discuss some of the critical social acceptance issues surrounding recommender applications in E-commerce including privacy and trust.

Technologists often assume that the "best" recommender application is one that is fully automatic and completely invisible. Our study does not bear this assumption out at all. We find many different recommender models, each of which is appropriate for a different set of business goals. Specifically, we reflect on the five models we presented above.

Broad recommendation lists are an effective way to leverage human experts and to provide community-wide recommendations for new customers or customers interested in branching away from their interests. Because they can be presented in so many ways, they give marketers many points of contact with customers and allow them to build relationships around regular editorial contact.

Customer comments and ratings can help sites supplement their credibility and create a greater sense of community. Reviewers are likely to visit the site each time they consume a product since they enjoy sharing their opinions and comment readers may come to depend on reviews to help guide their purchases.

Notification services help businesses serve content-focused customers by ensuring that they are quickly aware of interesting content. These customers view such notifications as a service, and the business builds a competitive advantage as it holds a profile of customer interests and proactively serves the customer.

Product-associated recommendations allow businesses to respond to each customer's current interests and allow the natural associations among different products to guide customers to the right purchase. These recommendations combine the helpfulness of a knowledgeable salesperson who can recommend items to match ones of interest with the layout of a good store where complementary items are conveniently shelved near each other, even if that means they are shelved in many locations. Customers appreciate good help and good organization and will return to a business that provides a pleasant shopping experience. They also make larger purchases when suggestive selling leads them to products they may have otherwise done without or bought later at another retailer.

Deep personalization emulates the relationship a customer has with an expert salesperson who not only knows the customer's tastes but also has the experience that comes from selling to many other customers. With deeply personal recommenders, businesses can identify and anticipate customer desires because they have indeed seen other similar customers before. The customers, in turn, discover that the more they shop, the better the store becomes.

Each of these models addresses different business needs that reflect different business models, customers, and marketing plans. By selecting the appropriate combination of recommender applications, businesses can maintain their competitive advantage, retain customers, and increase sales.

Appendix: Privacy issues in recommender applications

Privacy is an important issue in recommender applications. In order to provide personal recommendations, recommender systems must know something about the customers. Indeed, the more the recommender systems know, the better recommendations they can provide. Furthermore, E-commerce sites can learn a great deal about customers without the customers' awareness or consent. Customers are quite reasonably concerned about what information is collected, whether it is stored, and how it is used.

In the following, we consider aspects of the privacy issue of particular concern to E-commerce recommender applications. We first examine the types of personal data that customers may want to protect and that businesses may want to use. We then examine the issue of privacy policies and trusted brands and a social mechanism for ensuring privacy. Finally, we explore technological approaches for automating enforcement of privacy policies.

Personal data

Customers shopping at web sites today make extensive information available to the site:

- Explicitly provided preference information such as product ratings, comments, or registered attributes of interest.
- Implicitly provided preference information including the products and information viewed, time spent viewing, searches performed, items explored or placed in a shopping cart, and even the site from which the customer navigated to the E-commerce site.
- Transactional information when products are purchased including forms of payment, account numbers, shipping addresses, and products purchased and shipped to each address.
- Explicitly provided identification information such as name, address, e-mail address, and phone number.
- Implicitly provided identification information such as the IP address (and therefore often the name) of the machine or domain from which the customer is browsing.

In addition to this already-extensive set of information, many customers have begun to realize that a small set of identity information is sufficient for businesses to acquire extensive additional information from other businesses or data collection agencies. Accordingly, it is not surprising that customers want the ability to browse and even shop at a web site with some assurance that their information will not be used for nefarious purposes. Moreover, customers are quite wary of data being collected without their awareness or consent; they do not like the feeling of being monitored.

At the same time, however, businesses can make good use of this information. By learning more about a customer's preferences, they can provide extensive personalization. By learning customer demographic information such as ZIP codes, they can customize the site by selecting weather-appropriate products, featuring products that match local tastes, and displaying prices with actual shipping charges. Of course, by sharing customer information with partners, businesses can develop a greater collection of information and thereby better understand their customers.

Given this apparent conflict, it is therefore no surprise that many customers want businesses to promise that they will limit their use and storage of personal data. While some places have laws enforcing standard storage and use restrictions, most e-businesses instead provide these assurances through privacy policies.

Privacy policies

Privacy policies are statements by businesses that explain what information they collect and how they use it. A common element of privacy policies is a promise not to sell the information to other sites without user permission. Studies suggest that consumers are very concerned about the possibility of their information being shared by many sites. Other privacy policy promises include not using the customer's e-mail address for advertising and not calling the customer's phone number except in connection with an order.

Privacy policies are important to recommender systems because they often restrict the ability of the business to share their data with other businesses. The strongest privacy policies even limit the ability of the business to collect data about their customers at all, which makes personalized recommendations impossible.

The privacy policies on many web sites do not help alleviate consumer fears very much. Part of the reason is that these policies are often written in confusing legalese. Not only are the policies hard to understand, but they often reserve the right to change the policy at any later time, without notice, and may reserve other rights like the use of information for "business interests," which could broadly include selling private information for money or bartering it for other information. Another reason for the lack of confidence is the trend toward consolidation in the industry, which means that using personal information "within the company" may include sharing it with a variety of unexpected sites. Finally, it is difficult to determine whether privacy policies are actually being followed. Customers who shop at a web site and later receive junk e-mail may legitimately wonder whether the site violated its policy; such violations are extremely hard to prove.

Privacy policies will be more useful once they are standardized and once simple consumer-recognized representations are available. For instance, TRUSTe and BBBOnline are both working to create consumer brands that represent trust for E-commerce sites. These brands may create special logos that consumers learn to recognize as an assurance that the site meets a particular level of privacy protection. To date, however, no such trusted brand name has emerged, in part because of confusion on the part of E-commerce sites, in part because of confusion on the part of consumers, and in part because of missteps by the privacy brands themselves. TRUSTe, for instance, earned widespread disapproval for its slow reaction to Real.com secretly recording customer information and transmitting it to its servers while displaying the TRUSTe banner.

Indeed, it is the lack of a trusted brand name and the general difficulty of enforcing privacy policies that has led to the development of technological approaches to protecting privacy.

Technological approaches

There are two current technological directions for protecting privacy. The first direction assumes that the businesses cannot be trusted or audited and thus attempts to disguise or scramble personal information. The second direction attempts to automate the negotiation and enforcement of privacy policies.

There is a long history of "anonymizing" techniques for electronic communication. Hackers and whistle-blowers alike have learned to send messages through anonymizers or even to set up new e-mail boxes for one-time use. Some of these techniques can help protect E-commerce consumers as well. Customers can reject cookies to prevent sites from recognizing them on future visits. They can hide their true IP address by browsing through firewalls or proxies that aggregate many people behind a single address or by browsing through trusted anonymizers. Of course, these approaches have their limitations. The anonymizer itself must be trusted, lest it sell its mappings to the business. Also, E-commerce applications that require payment and delivery present two new problems. While digital cash has been used for some applications, it is still not widely accepted. Also, today's delivery services require an address to which merchandise can be shipped. It is conceivable that privacy concerns may result in the reemergence of digital cash or "single-use" credit card numbers and the creation of trusted delivery services that accept deliveries to a one-time pseudonym, but these services do not yet exist.

Anonymizing techniques are disasters for recommenders, because they make it impossible for the recommender to easily recognize the customer, limiting the ability even to collect data, much less to make accurate recommendations. If recommenders are to be successful in the long-term, alternatives must be developed that alleviate consumer concerns about privacy while maintaining the notion of persistent identity.

One such step in this direction is being taken by the Platform for Privacy Preferences (P3P) initiative of the World Wide Web Consortium (W3C). P3P is a protocol whereby a site creates a machine-readable version of its privacy policy in a format that makes it possible for computers to understand and negotiate about privacy. Customers entrust their private information to a privacy agent (possibly their web browser). When the customer visits a site, his P3P agent negotiates on his behalf with the site to learn the privacy policy. The agent then asks the customer which types of information he is willing to share with the site, given the privacy policy. Over time, the agent may learn the customer's preferences. For example, a customer may be willing to give her e-mail address or phone number if the site promises to use it only within a particular transaction. Customers benefit because they can enter information once and not have to re-type it, and they know that information will only be shared with sites that promise to use it only in ways the customer accepts. Sites benefit because customers are more likely to share information if they understand the privacy policy and are more likely to share information if they do not need to reenter it each time.

P3P also provides other mechanisms to help customers and sites create private relationships. The P3P agent establishes a unique cryptographic identity with each site, called a Pairwise Unique IDentifier (PUID). The PUID makes it more difficult for different sites to share information since each site knows the customer by a different PUID. Since many recommender systems can make recommendations based only on the users' actions at the site, the PUID may be all the site needs to make useful recommendations. Other recommenders require some personal information such as ZIP code or age. Customers may be more willing to share this information though a P3P relationship, secure in the knowledge that they can create a new PUID in the future to "disappear" from the site. Of course, this anonymity only provides protection through a purchase if the site agrees not to maintain a connection between the PUID and the customer's real identify as needed for payment and delivery.

The P3P solution has great advantages for recommenders. It provides a persistent identifier that the site can use to maintain data about a consumer while protecting the consumer by limiting the connection between the real-world identity and the site identifier. P3P also makes it easy for consumers to have multiple identities to represent their multiple different shopping modalities. For instance, a parent might choose to have a different identity while shopping for himself then when shopping for his children. This use of identity to indicate role would simplify the job of recommenders since they would not have to separately try to analyze role. If P3P is supported by the Web clients and embraced by consumers, it may open great opportunity for recommenders.

Privacy issues are a great challenge, and as yet no comprehensive solution exists. The social solution of privacy policies is appealing, but current policies are hard to trust. Trusted privacy seals-of-approval still have not emerged, and technological solutions can improve the privacy situation but still cannot avoid some need for trust, auditing, or both. Privacy in electronic commerce remains an active research area at the intersection of sociology, economics, and computer science. Successful privacy solutions are essential for recommenders, so consumers do not take the extreme approach of cloaking identity completely. The eventual solution is likely to be a combination of the social and technological solutions being developed today.

Acknowledgments

We gratefully acknowledge the painstaking efforts of the editors and anonymous reviewers in helping make this paper more appropriate for the data mining audience. We also appreciate the support and feedback of the GroupLens Research Group at the University of Minnesota. This work was supported by the National Science Foundation under grants IIS 9613960, IIS 9734442, and DGE 9554517. Support was also provided by Net Perceptions Inc., a company that Konstan and Riedl co-founded and that sells a recommender system for E-Commerce.

Note

1. The term "recommender systems" evolved to replace and broaden the use of the term "collaborative filtering" because the latter term refers to a specific algorithm for recommending. The term "recommender system" refers both to systems that specifically *recommend* lists of products and to those that help users evaluate products. Previously, these systems were distinguished as systems that provide "recommendations," those that provide "predictions" of user preference, and those that provide "community opinion." The recommender systems research community has embraced all three components.

References

Agrawal, R., Imielinski, T., and Swami, A. 1993. Mining association rules between sets of items in large databases. In Proceedings of ACM SIGMOD-93, pp. 207–216.

Avery, C., Resnick, P., and Zeckhauser, R. 1999. The market for evaluations. American Economic Review, 89(3):564–583.

Balabanovic, M. and Shoham, Y. 1997. Fab: Content-based, collaborative recommendation. Communications of the ACM, 40(3):66–72.

Basu, C., Hirsh, H., and Cohen, W. 1998. Recommendation as classification: Using social and content-based information in recommendation. In Proceedings of the 1998 National Conference on Artificial Intelligence (AAAI-98), pp. 714–720.

Breese, J., Heckerman, D., and Kadie, C. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI-98), pp. 43–52.

Good, N., Schafer, J.B., Konstan, J.A., Borchers, A., Sarwar, B., Herlocker, J., and Riedl, J. 1999. Combining collaborative filtering with personal agents for better recommendations. In Proceedings of AAAI-99, AAAI Press. pp. 439–446.

Herlocker, J., Konstan, J.A., Borchers, A., and Riedl, J. 1999. An algorithmic framework for performing collaborative filtering. In Proceedings of SIGIR'99, pp. 230–237.

Hill, W., Stead, L., Rosenstein, M., and Furnas, G. 1995. Recommending and evaluating choices in a virtual community of use. In Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems, pp. 194–201.

Konstan, J.A., Miller, B., Maltz, D., Herlocker, J., Gordon, L., and Riedl, J. 1997. GroupLens: Applying collaborative filtering to Usenet news. Communications of the ACM, 40(3):77–87.

Mani, D.R., Drew, J., Betz, A., and Datta, P. 1999. Statistics and data mining techniques for lifetime value modeling. In Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 94–103.

Peppers, D. and Rogers, M. 1997. The One to One Future: Building Relationships One Customer at a Time. Bantam Doubleday Dell Publishing, New York, NY.

Pine II, B.J. 1993. Mass Customization. Boston: Harvard Business School Press.

Pine II, B.J. and Gilmore, J.H. 1999. The Experience Economy. Boston: Harvard Business School Press.

Pine II, B.J., Peppers, D., and Rogers, M. 1995. Do you want to keep your customers forever? Harvard Business School Review, 1995(2):103–114.

Reichheld, F. 1993. Loyalty-based management. Harvard Business School Review, 1993(2):64–73.

Reichheld, F. and Sasser, W.E. 1990. Zero defections: Quality comes to services. Harvard Business School Review, 1990(5):105–111.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. 1994. Grouplens: An open architecture for collaborative filtering of netnews. In Proceedings of ACM CSCW'94 Conference on Computer-Supported Cooperative Work, pp. 175–186.

Salton, G. 1968. Automatic Information Organization and Retrieval, McGraw-Hill Book Company, New York, NY.

Sarwar, B., Konstan, J.A., Borchers, A., Herlocker, J., Miller, B., and Riedl, J. 1998. Using filtering agents to improve prediction quality in the grouplens research collaborative filtering system. In Proceedings of 1998 Conference on Computer Supported Collaborative Work.

Schafer, J.B., Konstan, J.A., and Riedl, J. 1999. Recommender Systems in E-Commerce. In ACM Conference on Electronic Commerce (EC-99), pp. 158–166.

Shardanand, U. and Maes, P. 1995. Social information filtering: Algorithms for automating "word of mouth." In Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems, pp. 210–217.

Shneiderman, B. 1997. Direct manipulation for comprehensible, predictable, and controllable user interfaces. In Proceedings of IUI97, 1997 International Conference on Intelligent User Interfaces, Orlando, FL, January 6–9, 1997, pp. 33–39.

Wolf, J., Aggarwal, C., Wu, K-L., and Yu, P. 1999. Horting hatches an egg: A new graph-theoretic approach to collaborative filtering. In Proceedings of ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, San Diego, CA.

J. Ben Schafer is completing his Ph.D. in the Department of Computer Science and Engineering at the University of Minnesota. Ben's research has included projects in agent-assisted interfaces, human-computer interaction, artificial intelligence, collaborative information filtering, and electronic commerce. His dissertation is entitled "MetaLens: Improving Recommender Systems for Electronic Commerce through the use of Multiple Data Sources."

Joseph A. Konstan is Associate Professor of Computer Science and Engineering at the University of Minnesota. Prof. Konstan holds a Ph.D. in Computer Science from the University of California, Berkeley, where he completed a dissertation on user interface toolkit technology. Since then, he has worked on a variety of human-computer interaction projects focused on collaborative information filtering, automation and authoring, multimedia, and visualization. Dr. Konstan is an ACM lecturer and currently serves as editor of the ACM SIGCHI Bulletin.

John Riedl has been a member of the faculty of the computer science department of the University of Minnesota since March 1990. His research is focussed on collaborative systems, which are computer systems that help groups of people work together more effectively. Since 1992 he has been co-directing the GroupLens Research project on collaborative information filtering. In 1996 he co-founded Net Perceptions, a leading one-to-one real-time marketing company. Riedl is Associate Professor at the University, and Director and Chief Scientist at Net Perceptions. Riedl received the B.S. degree in mathematics from the University of Notre Dame, Notre Dame, IN in 1983 and the M.S. and Ph.D. degrees in computer science from Purdue University, West Lafayette, IN in 1985 and 1990, respectively.