Recommender Systems

Jan Boehmer, Yumi Jung, and Rick Wash

Michigan State University

Email: boehmerj@msu.edu

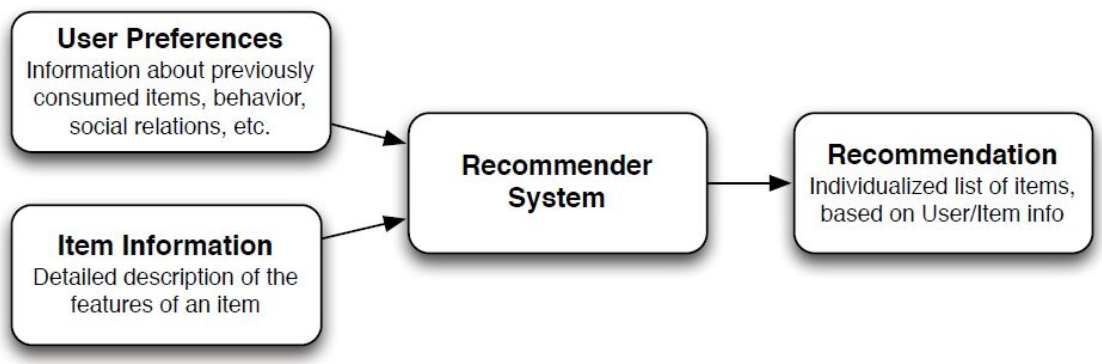
Word count: 4,744

Abstract – Recommender systems are a vital part of today's information society to deal with information overload, especially in e-commerce. Recommender systems help retailers to choose items to display based on customers' preferences, help users to search for items in personalized ways, and help streaming services create customized playlists. This entry describes multiple kinds of recommender systems and how they work. It also explains their historical and intellectual context, shows how they might affect users, and discusses current challenges.

INTRODUCTION

Information overload has become one of the major issues in today's information society. Modern communication technologies and especially the Internet have made it easier than ever to access information. However, the constant stream of information has often made it difficult for users to select the piece of information from this stream that is most helpful in a given situation. Recommender systems help to deal with information overload. They are a set of technologies used to sort information or goods a user might be interested in from the whole of an information source. For example, a commercial website like Amazon.com offers millions of different items such as books and movies which makes it impossible to display all of them to a user. When selecting a set of items to display to a given user, it is important to recognize that individual users have different preferences and the system therefore needs to identify specific items that are most likely to be relevant to each user. A recommender system does just that. It creates a list of items that are likely to be of interest to each individual user. Not only books or movies can be recommended, items can also be travel destinations, news articles, restaurants, friends within social networks, and even questions and answers. In sum, "any system that produces individualized recommendations as an output or has the effect of guiding the user in a personalized way" (Burke, 2002, p.331) can be defined as a recommender system. Recommendations occur not only when a user is actively searching for something, but also in a ubiquitous manner without a user's direct input. Targeted online advertising is also a form of recommender system.

Consider this practical example: Let's say Anne wants to buy a book in an online bookstore, but the store offers millions of titles, most of which Anne is not familiar with. Which book should she read? In a traditional bookstore, she might ask the bookseller she has known for years for advice. Online, a recommender system can help her. Instead of a general set of "featured items" that all users would see, it will automatically analyze the inventory of books and compare them to the information available about Anne (such as browsing and buying history, and stated preferences for previously read books). Anne tells the system (either explicitly or implicitly) which kinds of books she likes, and the recommender estimates a set of books that best match Anne's preferences and then provides her with a personalized view of the options, emphasizing the items that are most likely to be of interest to her. As a result, Anne receives her very own list of books she might like (see Figure 1).



Label as Figure 1: Diagram of the recommendation process

Many search engines are very similar to recommender systems. A search engine, such as Google, suggests the most relevant information to the user based on the user's queries. However, traditional search engines provide the same results for all users who type the same keyword. Some advanced search engines can be categorized as recommendation systems (Burke, 2002) because they provide different results to different users even if the users type the same keyword, which is a key factor of recommender systems. They provide individualized results by considering each individual user's characteristics such as past browsing histories and locations to provide relevant information to the one user who is looking for this information instead of providing results that are most likely to be relevant for the largest number of users.

While recommender systems got their start in traditional electronic commerce domains such as movies and books, they have been applied to a wider variety of contexts such as social media, news aggregating websites, and video streaming services. No matter which context they are applied in, however, recommender systems help individuals to make decisions or to access potentially more interesting information by providing personalized recommendations.

INTELLECTUAL & HISTORICAL CONTEXT

The invention of modern recommender systems dates back to 1992. At that time, electronic mail was becoming very popular in organizations and evolved as the most common medium used to share information among co-workers. As a result, employees were overwhelmed by the huge amount of incoming emails (Goldberg, Nichols, Oki, & Terry, 1992), and had a hard time filtering out the most relevant emails related to their tasks or preferences. The Xerox Palo Alto Research Center (PARC) invented Tapestry to address this issue, and in so doing, created one of the first recommender systems. Tapestry was designed to filter emails based on topics a user indicated as relevant. Those topic categories were not necessarily created by the program, but the users themselves could propose them based on their own interpretation and classification of the contents of the emails they receive. Other users could then use those topic categories to state their own preferences. In addition to filtering emails based on their content, Tapestry also estimated the importance of a message by assessing how "popular" an email was with other employees in the company who had received the same message. These two concepts, the recommendation of items based on their content and the popularity of those items among other people with similar interests, became the basis for modern content-based and collaborative filtering recommender systems. Content-based recommender systems filter items based on item similarities, while collaborative recommender systems sort items based on the similarities of users who have chosen similar items before.

Shortly after Tapestry was first created, the GroupLens research team led by Paul Resnick at MIT adapted the basic idea and developed a more applied recommender system for use within newsgroups. Newsgroups served as places of information exchange and discussion on the early Internet and became very popular. However, due to their increased popularity, soon there was too much information for any one person to read and discuss. The recommender system "GroupLens" was invented to assist users in finding relevant items in the large amount of content posted in newsgroups (Resnick et al., 1994).

GroupLens extended the basic idea of Tapestry in two ways. First, while Tapestry worked only within a given email server (for example within a particular company), the rating system for GroupLens' collaborative filtering was extended to multiple sites. This meant that GroupLens was able to add new newsgroups to its existing dataset so that the system could better predict user preferences for items. This led to an increase in the accuracy of recommendations. In addition, GroupLens was far more flexible than previous systems. It aggregated evaluations from different sources, recognized their patterns and created a new rating system. This enabled different rating systems to work together. Overall, the underlying principle and context of GroupLens is very similar to the recommender systems in use today and this is why it is widely considered to be the first modern recommender system.

Tapestry and GroupLens illustrate not only two early versions of recommender systems, but also the fact that both industry (Tapesty) and academia (GroupLens) have influenced each other in the development of recommender systems. Advances in recommender systems have often been the combined effort of scholarly research and commercial interests.

With the broad adoption of the World Wide Web during the early 2000s, and the early success of online retailing, the market opened up for large scale commercial applications. With the increase in items offered and many potential users to serve, for many commercial websites, such as Amazon, recommender systems became an important part of their businesses strategy. Their goal was to match user preferences with offered products to increase purchase rates. However, while it was commercial interests that brought recommender systems to public attention, it was academia that helped improve the systems. Neither was able to make significant process on its own since when researchers came up with new ideas to improve these systems, they needed large real-world data sets of products and individual preferences to empirically test the outcomes of their systems - data that only industry could provide.

One of the most prominent attempts to improve recommender systems was a collaboration of industry, academia, and private companies: the Netflix Prize. In 2006, Netflix, a provider of on-demand Internet streaming media, offered a prize of USD 1m to a person or group that could improve the existing Netflix recommender system by 10 percent. Some 20,000 teams from more than 150 countries registered for the competition. In the end, 2,000 teams submitted over 13,000 modifications to the existing system to predict users' preferences (Bennett & Lanning, 2007). The competition ran for three years and the participating teams tested their modifications to the Netflix system using the big dataset the company had provided. The dataset included anonymized information about users, movie descriptions, and ratings – and was highly interesting for researchers in academia. The Netfilx Prize offered scholars a unique opportunity to test their recommender systems developed in the laboratory using a real world dataset, which has not been possible before. In 2009, two teams reached that 10% mark almost simultaneously, but the competition was one by "BellKor's Pragmatic Chaos", a seven-person team of statisticians, machine- learning experts and computer engineers from the United States, Austria, Canada and Israel, some of whom worked for large technology companies.

TYPES OF RECOMMENDER SYSTEMS

Recommender systems make suggestions based on two types of data: background data and input data. Background data is the information that is already available in a system, independent of the system's current user. It is the corpus of books and their descriptions in a book recommender system, for example. This data is already available when a user joins the system. Input data is the information an individual enters into the system. The input data is transformed into background data so the background data is continuously updated. From these two sources the recommender system calculates the recommendations for each user (Burke, 2002).The three most common types of recommender systems are called content-based, collaborative filtering, and hybrid recommender algorithms.

Content-based recommendation

A content-based algorithm suggests items with similar properties to items the user has already liked or purchased. Based on the user's preferences, content-based recommender systems match the background information about the items stored in the database with the input provided by the user. This input might be in the form of explicitly stated preferences in the users' profile, or manifestations of those preferences such as browsing, and purchase behavior, or ratings given to items. The overall goal of the system is to recommend items that the individual has not yet seen but is likely to find interesting (Lops et al. 2011).

A book recommender system based on a content-based method, for example, would analyze the commonalities among the books an individual has purchased or rated highly in the past, such as specific authors, genres, and topics, and then recommend books that a have high degree of similarities to the individual's preferences. If a user has liked the first three "Harry Potter" books, or indicated J.K. Rowling as a favorite author, the recommender system might recommend the newest version of the "Harry Potter" book. Schafer et al. (1999) call this method "item-to-item correlation."

Collaborative recommendation

Collaborative recommender systems match the profile of the user who is looking for a recommendation with the profiles of other users in the same system with similar preferences. If other users have rated a set of items the same way as the target user, the recommender system predicts that the target user might also like new items which have not yet been seen, but that similar users have liked.

The underlying assumption in collaborative filtering is that the ratings an individual provides represent fairly constant opinions that can be collected and analyzed to provide a reasonable estimate of the actual preferences of the individual. Collaborative filtering systems aggregate ratings, recognize similarities between individuals, and generate recommendations based on inter-user comparisons (Ekstrand et al. 2011).

A book recommender system based on the collaborative recommender system, for example, finds other users who have exhibited a similar taste in books and the users who rated the same books similarly. Then, the system recommends books to the user that received a good rating from peers, but that the user has not read. This form of recommendation is also called “people-to-people correlation” (Schafer, Konstan and Riedl, 1999).

Collaborative filtering systems have several distinct advantages. First, they can be applied to almost any type of content. While content-based recommender systems can only analyze content that has at least some data associated with it that is readable by the system (i.e. textual information such as product descriptions), collaborative filtering also works well when the content is more complex (i.e. movies or music without descriptions).

A second advantage of collaborative recommendation is its simplicity. Rather than computing the potential numerical value for the utility of each item, collaborative filtering systems only focus on items rated highly by peers. By sorting items based on popularity, the recommender system can reduce its workload. Third, the collaborative approach is more likely to be able to provide cross-genre recommendations by taking advantage of the complexity of real user preferences.

Continuing the example above, if the user wants a movie to go with the book, the collaborative filtering system might be able to provide a recommendation based on the movies people with similar preferences have liked. A content-based system with information only about the user’s book preferences would only occasionally be able to suggest movies (i.e. when book and movie authors overlap).

Hybrid recommender systems

Hybrid recommender systems combine two or more recommendation methods to generate an estimate of the utility an item might have for a user. This is done mainly to use the advantages of one technique and address the disadvantages of the other. For example, a collaborative recommender system produces good recommendations based on preferences of similar users, but it cannot recommend new items that have not been rated by users. Since content-based recommender systems do not require users’ ratings but instead base their recommendation on the project descriptions that will most likely be available once an item is entered into the system, these can be used as a supplement in cases where insufficient rating data is available (Burke, 2002).

Different kinds of recommender systems can be combined to create a hybrid system. Sub-systems under a hybrid system can be merged into one hybrid recommender, or each method can be implemented independently and only their predictions combined to show recommendations. Most recommender systems are hybrid recommender systems such as the system employed by Amazon.com.

MAJOR RESEARCH CHALLENGES

While current recommender systems commonly produce high quality recommendations that match a user's preferences, there are two prominent challenges that are being addressed by researchers: the cold start problem and providing contextual recommendations. In addition, there is discussion about whether making recommendations changes user preferences.

Cold-start problem

The cold-start problem occurs when the system is asked to recommend items to new users who have not stated a preference on any item (lack of input data), or when the system encounters new items that no user has rated (lack of background data) (Lam, Vu, Le, & Duong, 2008; Ahn, 2008).

Since a recommender system uses the profiles of users and the descriptions of items stored in a system to predict preferences, a user who has not provided much information to the system creates problems. The cold-start problem is also called the “new user” problem and one way of overcoming this problem is by prompting the user to provide input data. This process might be facilitated by incentives to fill out a profile page or to provide a set of initial ratings for known items. Search queries or data imported from other web applications, such as social media sites, are further examples of helping the system to generate more input data.

Similarly, recommender systems also have to deal with new items. Frequently, there is very little information available on a new item because the item has not received a rating from a user or has no purchase history and it is difficult for the system to make accurate predictions. This is frequently called the “new item” or “early rater” problem and it is especially prevalent in collaborative filtering systems. Developers of such recommender systems try to address this problem by explicitly asking users to rate these new items.

Contextual recommendations

Most recommender systems assume that a user’s preferences are static so that each user’s preferences can be inferred from the sum of their past decisions. As an example, if a user in a movie recommender system has previously liked horror movies, the system will likely suggest another horror movie since it makes its recommendations based on the user’s previously stated preferences. Such a system focuses on recommending items without taking into account any additional information about the user’s context such as time, day, location and emotion – or the conditions under which the recommended item will be consumed.

However, preferences may change dramatically depending on the situation. The user might like the new “Harry Potter” movie when watching it at home alone, but when going out with friends, the user might strongly prefer the newest romantic comedy. In another case, a particular restaurant might really be a great fit given a user’s stated food preferences, but given the fact that the user has only an hour for his or her lunch break, driving across town to get to the restaurant might not be an option. Therefore, a less-than-ideal restaurant in a convenient location might provide a greater utility. Recommender systems are slowly incorporating context into their processes, for example, using location-based recommendations on smartphones that give more emphasis to recommending local items.

Another aspect of recommendation context is the part of a list or series of items that will be consumed in succession? Pandora, a presentation of the recommendations, Is the system recommending a stand-alone item, or will it be a music streaming service, for example, uses a recommender system to create individualized playlists of music instead of a single item.

This distinction has important implications for the recommendations the system has to make. If only one item will be consumed, the system needs to find the single best fit for the given set of preferences. However, if the system recommends a set of items that will be consumed as a combined product, the system has to make sure that the recommended items reflect the stated preferences and also ensure that they are sufficiently different to avoid being perceived as substitutes.

The system might be able to identify a user's favorite song, but listening to the same song over and over again might not add any further utility. Therefore, "accurate prediction of consumer preferences undoubtedly depends upon the degree to which the recommender system has incorporated the relevant contextual information into a recommendation method" (Adomavicius et al. 2005, 104).

Effects on Users

In addition to providing recommendations, recommender systems also have unintended effects on users. Cosley et al. (2003) have shown that when users were exposed to predicted ratings of items before they had rated items themselves, they changed their ratings in favor of the predicted value. Murray and Häubl (2005) also support the claim that recommender systems influence users' perceptions of recommended items. A user might think that, since an item is recommended based on stated preferences, that he or she should like that item. However, Lam and Riedl (2004, 394) argue that "whether this represents a genuine change in opinion is unknown - it might just be that users conform in what recommender systems say, not in what they believe".

Some users may doubt the recommendations presented by a system. A user might question whether the algorithm is producing recommendations based on the underlying data, or on behalf of a third party that is manipulating the results to sell a product. For example, a user in a book recommender system might wonder whether books from a certain publisher are preferred by the system, because the store might earn more by selling them. In this case, the user might doubt the system and not favor the recommendations – no matter how accurate they are.

Trust in the system – or lack thereof - has been shown to be an important determinant of whether users perceive a recommender system to be of high quality (e.g. Mobasher et al. 2007). Recommender systems often address this issue by making the underlying processes as transparent as possible. McSherry (2005) demonstrates the importance of transparency (user understanding how the recommendation is generated) in explaining the recommendations a particular algorithm makes and datasets on which they are based.

Overall, whether a user sees a recommender system as being useful depends on the accuracy of the recommendations and the perceptions of the user. Swearingen and Sinha (2001) have shown that user's perceptions of a recommender system are an important metric when assessing the quality of recommendations made by the system.

NEW CHALLENGES

The advent of social media and the increase in available data poses new opportunities, but also many challenges to recommender systems. First, social media produce very different types (posting, replying, sharing) or formats (text, link, pictures) of information. This is problematic, since it is difficult to transform this information into a format that is interpretable by the system. The information in social media is generated by its users which means that the meaning of the information is based on each user's interpretation and relationships. For example, information or a recommendation from user A might be meaningful to user B, but not to user C. This is a matter of trust or 'closeness' between users. Social relationships are based on subjective values that are hard to classify. While users are generating increasing amounts of information, they are doing so in a manner that makes it difficult to use for making recommendations.

The most popular form in which such information is generated is tagging (Burke et al. 2011). Users can tag diverse items such as people, applications, and pictures using diverse forms of expression such as liking and linking. This is difficult to standardize because the meaning of actions not only differs among the same actions across different platforms, but more importantly, the meaning also differs in the perceptions of people expressing them. For example, a "like" on a social networking site might mean "this is the best item that I have ever seen" for one person, but it might mean "this item is not bad" for another person. A recommender system cannot differentiate between the two meanings of "like", and would assign both of them the same interpretation, causing inaccurate results.

Another issue is privacy. Recommender systems operate in a social environment, often use user-generated content as input data, and operate mostly on personal information. Users may be anxious as to whether their personal information can be seen or inferred by others. Although a recommender system incorporates information from users, a certain level of obscurity may be expected by the user. Obscurity is an important part of privacy which can be maintained by low search visibility (Hartzog and Stutzman, 2010). Any system that uses personal information and exposes this information to others reduces the level of obscurity by reorganizing personal information in ways the user does not necessarily intend. It is unclear how this privacy concern affects recommender systems and how it can be addressed by modifying the systems to reduce the concerns.

Recommender systems are becoming increasingly important in today's information society. They are shaping the way individuals are consuming information. Recommender systems are ubiquitous on the Internet, and users often do not realize that their decisions are at least partially driven by algorithms. Understanding the ways in which recommender systems work and how they might influence perceptions and behaviors is important – it will become more so in a fast paced information society.

See also:

Electronic Commerce Reputation Systems

Electronic Commerce and Online Trust

E-Commerce Business Models

Location-Based Commercial Services

AUTHORS:

Jan Boehmer is a PhD student at Michigan State University in the School of Journalism. He investigates how new and social media technologies affect users' perceptions and behaviors of online content, focusing on interactivity, civic participation, and engagement in the realms of journalism and public relations. He holds a degree in Online Journalism from the University of Applied Sciences in Darmstadt, Germany, and has worked as a writer, editor, and social media manager for radio, TV, newspaper, and online media.

Yumi Jung is a PhD student at Michigan State University in the Department of Telecommunications, Information Studies, and Media. Her work involves understanding how people perceive their privacy issues in the context of social relationships, and what the benefits of social relationships within social media systems are. Jung completed her master's degree in Communication from Rutgers University and her bachelor's degree in Sociology from Hanyang University in South Korea.

Rick Wash is an Assistant Professor at Michigan State University in the School of

Journalism and the Department of Telecommunications, Information Studies, and Media. His work involves understanding how people think about their interactions with computers, and their interactions with other people through computers, with a particular focus on security and collaborative systems. He completed his PhD at the School of Information at the University of Michigan. Prior to studying information, Rick completed his master's degree in Computer Science from the University of Michigan, and his bachelor's degree in Computer Science from Case Western Reserve University

REFERENCES:

Adomavicius, G., Sankaranarayanan, R., Sen, S., & Tuzhilin, A. (2005). Incorporating contextual information in recommender systems using amultidimensional approach. *ACM Transactions on Information System (TOIS)*,*23*(1), 103-145.

Ahn, H. J. (2008). A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. *Information Sciences*, *178*(1), 37-51.

Bennett, J., & Lanning, S. (2007). The Netflix prize. Proceedings from: *KDD cup and workshop '07*, San Jose, CA: ACM (pp. 35 – 39).

Burke, R. (2002). Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction 12*(4): 331– 370.

Burke, R., Felfernig, A., & Göker, M. H. (2011). Recommender systems: An o verview. *AI Magazine, 32*(3), 13-18.

Cosley, D., Lam, S.K., Albert, I., Konstan, J., & Riedl, J. (2003). Is Seeing B elieving? How Recommender Systems Influence Users' Opinions. In *Proceedings of CHI 2003 Conference on Human Factors in Computing Systems*, Fort Lauderdale, FL: ACM, (pp. 585-592).

Ekstrand, M. D., Ludwig, M., Konstan, J. A., & Riedl, J. T. (2011). Rethinking the recommender research ecosystem: reproducibility, openness, and lenskit. In *Proceedings of the fifth ACM Conference on Recommender Systems* (pp. 133-140), Chicago, IL: ACM

Goldberg, D., Nichols, D., Oki, B, & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM 35*(12), 61 – 70.

Gretzel, U., & Fesenmaier, D. R. (2006). Persuasion in recommender systems. *International Journal of Electronic Commerce*, *11*(2), 81-100.

Lam, S. K., & Riedl, J. (2004). Shilling recommender systems for fun and profit. In *Proceedings of the 13th international conference on World Wide Web (WWW2004)* (pp. 393-402). New York, NY: ACM

Lam, X. N., Vu, T., Le, T. D., & Duong, A. D. (2008). Addressing cold-start problem in recommendation systems. In *Proceedings of the 2nd International Conference on Ubiquitous Information Management and Communication (ICUIMC)* (pp. 208-211), Suwon, Korea: ACM.

Lops, P., Gemmis, M., & Semeraro, G. (2011). Content-based recommender systems: State of the art and trends. In F. Ricci, L. Rokach, B. Shapira, &

P.B. Kantor (Eds.), *Recommender Systems Handbook* (pp. 73 – 105), New York, NY: Springer.
McSherry, D. (2005). Explanation in recommender systems. *Artificial Intelligence Review*, *24*(2), 179 - 197.

Mobasher, B., Burke, R., Bhaumik, R., & Williams, C. (2007). Toward trustworthy recommender systems: An analysis of attack models and algorithm robustness. *ACM Transactions on Internet Technology (TOIT), 7*(4), 23.
Murray, K.B., & Häubl, G. (2005). Processes of preference construction in agent-assisted online shopping. In C.P. Haugtvedt, K.A. Machleit, and R.E Yalch (eds.), *Online Consumer Psychology* (pp. 265–283), Mahwah, NJ: Erlbaum.
Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). GroupLens: an open architecture for collaborative filtering of netnews. In Proceedings of the ‘94: *ACM conference on Computer Supported Cooperative Work (CSCW)* (pp. 175-186). ACM.
Schafer, J. B., Konstan, J., & Riedl, J. (1999). Recommender systems in e-commerce. In *Proceedings of the 1st ACM Conference on Electronic Commerce* (pp. 158-166). Denver, CO: ACM.
Swearingen, K., & Sinha, R. (2001, September). Beyond algorithms: An HCI perspective on recommender systems. In *Proceedings of the ACM SIGIR’01 Workshop on Recommender Systems,* New Orleans, LA: ACM.