Knowledge Based Recommender Systems Using Explicit User Models

Brendon Towle & Clark Quinn
Knowledge Planet.com
5900 Hollis Street, Suite A
Emeryville, CA 94608
{btowle, cquinn}@knowledgeplanet.com

Abstract

While recommender systems are in widespread use, they still experience problems. Many recommender systems produce recommendations which the customers find unsatisfactory. Further, these systems often suffer from problems when there are not enough participants, or when new products enter the system. We perceive an opportunity for knowledge-based recommender systems to gain leverage on recommendation tasks by using explicit models of both the user of the system and the products being recommeded. This differs from previous systems which, when they use a user model, have used one that is inferred from the ratings given by that user (i.e., an *implicit* user model). We believe that the additional information given by the user and product models can give the system leverage in difficult recommendation tasks, and also alleviate both the "early rater" problem and the "sparse ratings" problem experienced by current recommender systems.

Introduction

One of the opportunities of including technology into buyer-seller relationships is using data storage and processing capabilities to add perceived value to the interaction. The internet (and the resulting explosion in e-commerce) has provided a technology-mediated marketplace and consequently provided opportunities to leverage interactions between buyers and sellers through technology design.

One of these opportunities is using information on the particular buyer, and information on the objects purchased by that buyer, to do more specific mapping between purchaser and potential purchases. Systems can use information about the purchaser such as past purchases, demographics, specific inquiry patterns, to infer things about this inquiry for purchase. Systems can use information about properties of available items to prioritize such items to meet the inferred (or explicit) profile of the purchaser. In essence, the system can 'recommend' items to individuals.

The Opportunity

Such a system can work in several ways: proactively before, or during, active shopping; based upon existing information about the shopper, through a shopper-initiated interaction, or through a system-initiated interaction. The fundamental issue here is the potential mechanisms and sources of data upon which to generate a recommendation. What we have, however, is an opportunity for systems to provide individualized suggestions for marketplace products.

On the surface, this may not seem like much of an opportunity; many online merchants incorporate some sort of recommender system, and research into recommender systems has been fruitful since the early 90s; see (Resnick et al. 1994) for a relatively early example. Two main types exist: statistical approaches that look at aggregate behavior of previous customers to make future recommendations, and knowledge-based approaches that do an explicit model of appropriate behavior. Each has drawbacks: statistical approaches require large amounts of initial data and can't handle certain types of relationships, while the knowledge-based approaches require deep engineering (Burke 1999). Combining these approaches, as they stand now, is still an open research question. Further, we see three challenges that stand in the way of large-scale success of recommender systems, even those that incorporate the best features of both the current research directions.

The First Challenge: Getting It Right

Anecdotal experience suggests that customers ignore recommendations after a while if they aren't consistently good. For example, one of the authors spent a fair amount of time entering data into both the Alexandria Digital Library (http://www.alexlit.com/) and CDNow (http://www.cdnow.com/), giving up when both of those systems failed to provide recommendations which he found useful.

To aggravate this problem, there are a number of systemic reasons why collaborative filtering systems might fail to produce useful recommendations:

- Data-driven systems, such as those used by Amazon, etc., fail to capture the fact that two different raters can

like the same product for different reasons, or that one rater can like two divergent products for different reasons. These different reasons then give different implications for both what should be recommended next to those raters, and for how those raters' information should be used to influence recommendations to others; neither of these facts are captured by current data-driven systems.
- The systems used by Amazon etc. are fundamentally *preference-based* systems; they address user's preferences for material. While useful, this fails to capture situations where users can be said to have *needs* for material; situations such as education, where one user might need a different type of course than another.

There is a clear relationship between the perceived quality of the recommendation and the likelihood of accepting the recommendation. There is also a tradeoff between the amount of activity required on the part of the customer and the utility of the recommendation. To the extent that automatic accumulation of data, and a sound basis for recommendation happen, there is a possibility that the recommendation will be accepted.

The Second Challenge: New Items Constantly Arrive

In many potential marketplaces (CDs, movies, books, cars, courseware, etc) there is a constant flow of new items into the market. This means that, at least for these items, the standard data-driven solution won't work, because there is no data to leverage; this is an instance of the 'early rater' problem, as described in, for example (Gokhale & Claypool 1999).

From a business perspective, this may not be much of a problem if the new items are CDs priced at $15US each. However, if the new items are electronic courseware priced at $15,000US each, the lack of recommendations for the new items becomes much more problematic, especially if there is a feedback cycle between recommendations and ratings: unrated items are less likely to be purchased, and thus less likely to be rated.

The Third Challenge: Some Products Have Different Characteristics

In his keynote talk at the Recommender Systems workshop at SIGIR'99, Konstan presented four characteristics of a product that make it easy to write a recommender system for that media:

- The product is targeted at a particular audience;
- People will tolerate inaccuracies in the recommendation, because of the low investment required;
- There is a lack of temporal decay in the relevance of the product;
- There is a lack of *portfolio effects*, where products that the user has already seen or purchased affect the interest in other products of similar description.

These characteristics are more or less true of movies, books, and music; all of the typical domains for recommender systems. Research done in the domain of news recommendations (a domain with substantial potential for portfolio effects) has tended to ignore the possibility of such effects; however, the potentially detrimental effects of reading a second news story on a subject are fairly small.

However, take the case of recommending courseware. There are substantial portfolio effects for such recommendations (if you've already taken a course in MSWord, no other MSWord course is likely to interest you), and the tolerance for inaccuracies in the recommendation is much lower. There is greater investment of time and money required, and consequently there are greater consequences from getting a bad recommendation.

These three challenges indicate that a more general solution for recommendation is needed.

The Solution

We proposed a mixed mode solution to this problem, based around explicit models of both the objects in the marketplace and of the customers, along with an intelligent system which performs a mapping between the two. Burke (1999) similarly proposed a hybrid system with explicit models of the objects in the marketplace, however his approach focuses on allowing users to critique the recommendations made by the system, and aggregating those critiques. Our focus is more on explicit models (of both content and users/customers) and aggregated behaviors. Further, Burke's approach focuses on allowing the system to answer the question "What can I buy that is like X?", while our approach focuses on allowing the system to answer the question "What can I buy that I might like/need?"

Modeling of Objects in the Marketplace

We propose, as does Burke, having explicit models of each product in the marketplace. For example, a story about the Whitewater scandal would be explicitly labeled with "political scandals" as a component of its content. This differs from previous content-based recommender systems such as P-Tango (Claypool et al 1999) or NewsDude (Billsus 1999), which use feature extraction as a means to develop a model of the content. However, feature extraction is essentially limited to content which is textual in nature; images or sound clips are not yet amenable to automated feature extraction, thus limiting the utility of this as a means of modeling content. This does require initial tagging of the marketplace objects, but we believe that will become part of the product origination process.

Modeling of Customers in the Market

Additionally, we propose having an explicit model of each customer in the market. Depending on the nature of the market and the requirements for accurate sales, this model may be derived from active querying of the customer, or from historical data of previous customer interactions. Further, this model might or might not share vocabulary with the model of the objects in the marketplace. We believe that the existence of a customer model along with the product model will alleviate the special case of the early rater problem that occurs when a new product enters the marketplace; since both customers and products are modeled, the new product can be recommended to customers.

Using explicit models of the people in addition to the standard correlations has been tried before, in the context of a recommender system using a Bayesian mixed-effects model (Condliff et al 1999), and with only modest benefits over standard collaborative filtering. Our proposal differs from their work in two important respects. First, we propose that the model explicitly include an indication of the causality with respect to the media in question. Second, we suggest that the dimensions upon which the customers are modeled be chosen for their perceived causality, as opposed to the convenience of gathering the demographic data. [Condliff et al. acknowledge that the features they chose may not have been particularly informative; one system used age, gender, race and high school attended to try to predict beverage preferences, and the other used age and US region from the EachMovie dataset (McJones 1997) to try to predict movie preferences.]

Knowledge-Based Systems Which Map Between the Two

Given these two models, a knowledge-based system can be built which performs a mapping between the two, associating (recommending) certain objects in the marketplace with certain customers in the marketplace. The nature of the domain may well determine both the syntax and the semantics of this intelligent system. For domains that have strong consequences for bad recommendations, a system based in strong research may well be imperative. For domains without strong consequences, a system based on anecdotal experience or models may be sufficient. In either case, we can expect there to be a fair amount of knowledge engineering involved in creating the intelligent system.

Addition of Data-Driven Component

In addition to the model-based recommendation generation described above, a standard data-driven component can be used to cover cases where the model is inaccurate or incomplete. However, the data-driven component of the system can be extended to use correlations between a user and the components of the model, as opposed to merely using correlations between users.

This gives the eventual recommendation algorithm three sources of information to use in making recommendations: the rating predicted by the user and content models, the rating predicted by computing correlations between the user and the features of the content, and the rating predicted by computing correlations between users. The exact manner of combining these sources is currently underspecified, but is probably unimportant. For example, Claypool et al (1999) describe a system which uses two sources of information, and combines them with a weighted average, adjusting the weights to minimize error. Alternatively, Pazanni (in press) describes a system that simply assigns points to items based on their ranking in the recommendations given by different methods, and adds up the points to produce a final recommendation. Both of these authors found their methods to perform adequately, suggesting that either of these methods, or a third method, would probably be adequate.

Example: Book Recommendations

As a first example of how this might work in practice, we present the experience of one of the authors. Books present one end of the spectrum, being a low-cost, low-portfolio-effect item. There are large amounts of purchases and consequently similarly should be lots of data useful for establishing preferences. Here is the experience of one of the authors getting book recommendations from a (human) recommender:

> I was at a bookstore that I frequented during grad school, looking for books. In my years, I had gotten to know one of the employees, and so I asked for recommendations. After asking me to remind him of the books that I liked and disliked, he asked me to choose between a false dichotomy: did I read for plot or for style? I told him that I read mainly for plot, but appreciated style as well. He came back with several recommendations, and I purchased two: one that he said was largely a style-based recommendation, but that might appeal to me, and another that he said was mainly plot-based, but with excellent style. The style-based recommendation was one that I found to be merely so-so, while the plot-based recommendation was a book that I consider to be the best fiction book I've read in the past 5 years.

This person's method of recommending books clearly incorporates methods that would be modeled by the standard collaborative filtering, since he asked for a list of data about the author's preferences (implicitly aligning them with previous experiences, and therefore incorporating a customer model). However, his recommendation algorithm also clearly incorporates a content model, as evidenced by the plot-vs-style question. And, this recommendation algorithm is clearly quite accurate, as the plot-based recommendation was exactly right. Feedback from the author would also clearly update his model.

Example: Electronic Equipment

On the other hand, consider the situation for high-cost, high-portfolio-effect commodity. The other author had the experience of purchasing a video camcorder. Individual purchasers seldom purchase another when an adequate one is owned, so the standard recommender system algorithm (if you liked the Sony, you'll probably also like the Panasonic) does not have the data required. However, there are a wide variety of features of both the camera and the user that can be used to match customers to products.

In this case, the salesperson's initial inquiries were aligned around features necessary, as a mechanism to distinguish between competitive offerings. This is a content model. This helped the salesperson narrow down to a particular model from each of several manufacturers. The salesperson then indicated information about the popularity of the particular manufacturers, augmenting the feature-based information with collaborative filtering. Implicitly, this characterized the types of customers that choose the different manufacturers (say: people for whom quality is more important than price might choose a brand known for quality, while those who care more about value tend to choose a brand known for price), adding a customer model to augment the feature or product model.

Conclusion

We believe, as in the examples, that combining customer models with product models, and using rules to map between them provide a richer picture than either alone.

Recommender systems can be generalized from their existing formats to make a more universal system. The requirements are to develop mechanisms that parallel the strengths of statistical data with explicit conceptual frameworks. While there are several ways to do this, we argue that in certain domains (if not most domains) strong models of individuals and objects provide a mechanism for one approach.

We are developing a system that acts as a recommender in a domain where redundant recommendations are intolerable. Our needs are to have very precisely defined recommendations. Consequently, we have determined that a very rich model of the customer, coupled with a similarly rich model of the commodity, and linked through a rich set of rules that capture relationships between customer characteristics and appropriate objects, provides us with a solution. We are in the midst of implementation, but expect to have a prototype in operation during Q2 of 2000. In the interim, we can point to the directions we're taking.

References

Billsus, D., and Pazzani, M. 1999. A Personal News Agent that Talks, Learns and Explains. In *Proceedings of the Third International Conference on Autonomous Agents (Agents '99)*, Seattle, Washington.

Burke, R. 1999. Integrating Knowledge-based and Collaborative-filtering Recommender Systems. In *Proceedings of the Workshop on AI and Electronic Commerce.* AAAI 99. Orlando, Florida

Claypool, M.; Gokhale, A.; Miranda, T.; Murnikov, P.; Netes, D.; and Sartin, M. 1999. Combining Content-Based and Collaborative Filters in an Online Newspaper. In *Proceedings of Workshop on Recommender Systems: Algorithms and Evaluation.* Berkeley, California

Condliff, M K.; Lewis, D. D.; Madigan, D.; and Posse, C. 1999. Bayesian Mixed-Effects Models for Recommender Systems. In *Proceedings of Workshop on Recommender Systems: Algorithms and Evaluation.* Berkeley, California

Gokhale, A., and Claypool, M. 1999. Thresholds for More Accurate Collaborative Filtering. In *Proceedings of the IASTED International Conference on Artificial Intelligence and Soft Computing.* Honolulu, Hawaii

Konstan, J. 1999. Recommender System Research: Perspective and Thoughts. Keynote Address, *Workshop on Recommender Systems: Algorithms and Evaluation*. Berkeley, California.

McJones, P. (1997) EachMovie collaborative filtering dataset. DEC Systems Research Center, http://www.research.digital.com/src/eachmovie.

Pazzani, M. (in press). A Framework for Collaborative, Content-Based and Demographic Filtering. *Artificial Intelligence Review*

Resnick, P.; Iacovou, N.; Sushak, M.; Bergstrom, P.; and Riedl, J. 1994. Grouplens: An Open Architecture for Collaborative Filtering of Netnews. In *Proceedings of Computer Supported Collaborative Work Conference 1994.* Research Triangle Park, North Carolina