0810-9-1

Internet Recommendation Systems

Asim Ansari[1]
Skander Essegaier
Rajeev Kohli

Columbia University

July 1999

[1]Please address all correspondence to Asim Ansari, 517 Uris Hall, Columbia University, 3022 Broadway, NY, NY, 10027

Abstract

A number of online firms including *Yahoo!*, *Amazon.com* and *Movie Critic* recommend documents and products to consumers. Typically, the recommendations are based on content- and/or collaborative filtering methods. We examine the merits of these methods, suggest that preference models used in marketing offer good alternatives, and describe a Bayesian preference model that allows statistical integration of four types of information useful for making recommendations: a person's expressed preferences, preferences of other consumers and experts, item characteristics and individual characteristics. The proposed method accounts not only for preference heterogeneity across users, but also for unobserved movie heterogeneity by introducing the interaction of unobserved product attributes with customer characteristics. Estimation using Markov Chain Monte Carlo methods is described. The model is used with a large data set, obtained from the Internet, to recommend movies under conditions in which collaborative filtering methods are viable alternatives, as well as under conditions in which no recommendations can be made by these methods.

Keywords: Software Agents, Internet User Modeling, Database Marketing, Mass Customization, Movie Preferences, Hierarchical Bayes, MCMC methods.

1 Introduction

Recommendation systems provide a type of mass customization that is becoming increasingly popular on the Internet. Search engines like *Yahoo!* and *Alta Vista* use them to recommend relevant documents based on user-supplied keywords. Other companies, like *Los Angeles Times,* customize their online versions based on customer interests. Still others recommend books, restaurants, music and movies, based on the likes and dislikes of other people with similar tastes. The ostensible value of such customization is that it decreases search effort for users and rewards a firm with greater customer loyalty, higher sales and more advertising revenues. And as a company learns more about its customers, it can deliver more targeted promotions and advertisements, both for its own products and for other products targeted to similar audiences.

Collaborative filtering is the earliest and best known of the current stock of customization techniques. It is used in the popular recommendation systems marketed by *Firefly* and *Net Perceptions.* In its simplest implementation, the method expresses a person's preferences as a linear, weighted combination of other people's preferences. The weights are proportional to correlations between the expressed preferences of a person and of all other people whose preferences are known for at least two common stimuli (e.g., movies). Commercial implementations of collaborative filtering incorporate proprietary enhancements to deal with problems that arise due to the use of non-statistical estimation methods. More recently, statistical methods – mixture (latent class) models and Bayesian networks – have been examined as alternative collaborative filtering methods. An early assessment of these methods by Breese et al. (1998) is not encouraging, their performance on reported predictive criteria being worse than that of simple collaborative filtering. On the other hand, the predictive accuracy of even the best reported collaborative filtering implementation suffers drastically when the database is sparse i.e., when the number observations per person in the database is small. And as best as we can tell, there are no published reports comparing the recommendations of these customization methods with the *uncustomized* recommendations that might be obtained from such aggregate feature based models as a pooled regression of preference scores on product features and/or expert recommendations.

This paper develops a hierarchical Bayesian model for recommendation systems. We test the model using data from an actual recommendation system for movies. Our model differs from existing recommendation systems in at least three respects. First, it simultaneously incorporates five different types of "information" : (i) a person's expressed preferences, (ii) preferences of other consumers, (iii) expert recommendations, (iv) item characteristics and (v) individual demographics. In contrast, available recommendation systems use one or two pieces of information, typically a measure of similarity between a pair of vectors representing preferences in collaborative filtering and word frequencies in content filtering.[1] Second, in our approach, nested models can be estimated that allow predictions when only certain types of information, such as demographics and expert ratings, are available for a person or a product. One can thus make recommendations even if no preference information is available for a person, and can incorporate incrementally expressed preferences to make better predictions. Third, despite the admitted value of having "dynamic, learning" recommendation systems, none of the existing recommendation systems of which we are aware incorporate much learning except in the most obvious sense that as more observations become available, the above mentioned similarity measures can be re-computed using a larger set of data. In contrast, as the present model is estimated using Bayesian techniques, it can incorporate learning in a more natural way. This can be implemented by combining quick-and-dirty online updating with periodic off-line updating.

Technically, our method treats both individuals and items as random samples from relevant populations. Like the hierarchical models by Allenby, Arora and Ginter (1995), and Rossi, McCulloch and Allenby (1996), it allows for unobserved heterogeneity in consumer preferences. In addition, it introduces unobserved product heterogeneity, which accounts for unobserved product characteristics that represent aspects of "holistic" customer evaluations. Our method, therefore, simultaneously allows for heterogeneity in customer preference structures and product appeal structures.

[1]Shardanad and Maes (1995) and Balabanovic and Shoham (1997) provide examples of systems that combine content and collaborative filtering. For example, Balabanovic and Shoham compares attribute-based profiles of users to recommend items with high attribute-based and other-users' preference scores. Sarwar et al. (1998) use filtering agents (filterbots) that act like normal users in a collaborative filtering system and return ratings on articles based on certain semantic information.

The rest of the paper is as follow. In the next section we briefly discuss recommendation systems. In Section 3 we describe our set of models and their estimation using Markov Chain Monte Carlo methods. In Section 4 we test our models using a large data base on movie preferences and compare their performance. In Section 5 we conclude by discussing future directions.

2 An Overview of Recommendation Systems

While Internet companies like *Firefly, Amazon.com* and *Yahoo!* have popularized "intelligent agents," the concept dates back at least to Negroponte (1970) and Kay (1984, 1990). Nor is the use of such agents limited to the Internet, as the following early, unsuccessful examples illustrate.

For a short time, *Blockbuster video* introduced in-store kiosks recommending films based on a member's past rental history (West et. al. 1999). This made possible such interesting recommendations as a pornographic film to kids and *Teletubbies* to grandparents living in the same household. Its effect on family well being are not known. Then there was *Magnet* (Levy 1993), which claimed to be the "first intelligent agent for the Macintosh." Essentially a file manager, the "agent" silently threw everything it found in the trash when a user mis-typed the name of a destination folder for some files (Foner 1993).

Recommendation systems are agents of the sort used by Blockbuster: they make suggestions to users, based on their past behavior and/or expressed preferences. Most recommendation systems use some sort of *filtering* algorithm, which in marketing parlance are individual preference models. Search engines that retrieve documents based on keywords are an example of a *content-based* system. In one commonly-used system, a document in which a keyword appears more often is considered more relevant. Frequency of word matches is also used to assess document similarity, typically based on the inner product (cosine) of the word vectors representing two documents. Sometimes, semantic content is used instead of words (Latent Semantic Indexing) and genetic algorithms are used to update user profiles that are used for screening documents (Salton and Buckley 1988).

As in applications of conjoint analysis for new product design, recommendation systems can be useful for reducing the search space of alternatives. However, there are two differences. First, in product design, the objective is to identify promising new products, where as recommendation systems suggest items that already exist. In other words, one seeks a common set of items (products) to offer to all people in one case, and unique items for each person in the other. Second, a company's objective in product design is to optimize some market performance measure across people. In selecting a recommendation system, one wishes to make the best possible dis-aggregate predictions for each person.

Another difference between product design and recommendation system applications is that while the space of products searched by optimization algorithms is typically large in product design applications, it can be small for recommendation systems. While the literature on recommendation systems emphasizes filtering applications (Maes 1994), the use of recommendation systems when there are only a few available alternatives is likely to become more important. To illustrate, suppose you are interested in seeing a movie (or a musical performance or a play) this weekend. Suppose also that you live in a small town, or that you only want to go to a neighborhood theater. You therefore have a few choices, but may still want to know which, if any, performances are worthwhile. As the example illustrates, choices among such *experience goods* – neighborhood restaurants, dry-cleaners, plumbers, – or among *reputation goods* – like physician and pharmacists, lawyers and legal advisors, financial institutions and real estate brokers — are facilitated by recommendations from others, who sometimes are friends and at other times are trusted experts like certain film critics or doctors. Indeed, it is useful to incorporate recommendations from others even for choices among *search goods,* because people do not always have the means for making such judgements as car safety or the expertise for evaluating "NON ECC SD RAM (1 DIMM)," an option with Dell's online "reconfigurator."

Finally, the ability to work with little individual-level information is more important for recommendation systems than for other marketing problems. In traditional market research studies, a respondent is either compensated for participating in a one-time study or choice data are available from customer panels over a relatively long period of time. In contrast, the folklore for online recommendation systems is that most people are averse to answering too

many questions before they start getting recommendations. This is more likely in such low-risk contexts as movie recommendations, especially as a user is unsure of the quality of the subsequent recommendations. In our application, for example, we find that while the number of movies rated by at least one person is very large, each person rates only a few movies.

The best known method for customizing recommendations is *collaborative filtering.* It is therefore useful if we examine the method in some detail before proceeding to a description of our proposed model.

2.1 Collaborative Filtering

In principle, this method mimics word-of-mouth recommendations by and for people with similar preferences. Typically, the method uses self-reported ratings over subsets of well-defined alternatives. First introduced by Goldberg et. al. (1992), the best-known commercial collaborative filtering algorithms are offered by *Firefly* (used by *Yahoo!* and *Barnes and Noble online*) and *Net Perceptions* (used by *Bertelsmann* and *Ticketmaster*). LA Times, London Times, CRAYON and Tango [MSNM99] also use collaborative filtering to customize online newspapers; Bostondine uses it to recommend restaurants in and around Boston; Sepia Video Guide uses it to make customized video recommendations; and Movie Critic, moviefinder and Morse use it to recommend movies.

There are two general classes of collaborative filtering algorithms. Memory-based algorithms operate over the entire user database to make predictions. Model-based collaborative filtering, in contrast, uses the user database to estimate or learn a model, which is then used for predictions. Model-based methods are a recent addition to the literature. We refer the reader to Breese et al. (1998) for a description of the latter, which are recent developments still under testing, and which use Bayesian networks (Heckerman, 1996) and finite mixture models (Chien and George 1999).

In the most common version of memory-based collaborative filtering systems, person $i's$ preference for a stimulus j, p_{ij}, is predicted using previous information provided by person i, the preference ratings of other persons, and a set of weights that are computed from the

database of preference ratings. The predicted rating is given by

$$p_{ij} = m_i + \sum_{k=1}^{I} w_{ik}(r_{kj} - m_k) \tag{1}$$

where m_k represents the mean rating of person k, computed across all stimuli rated by the person, w_{ik} is a normalized weight that reflects similarity or correlation between user i and each user k, and r_{kj} is the rating provided by person k, $k \neq i$, for stimulus j. Typically, the weight w_{ik} is some measure of similarity between the ratings of two users (correlation coefficient) and is computed by considering only the items simultaneously rated by the users i and k.[2] The predicted rating in Equation (1) is a weighted average of the ratings provided by other users. In computing the predictions, a greater weight is placed on the ratings of those users who have similar preferences to the target user.

Collaborative filtering algorithms have several limitations. First, when data are sparse, the correlations (weights) are based on very few common items and hence are unreliable. Breese et al. (1998) show that prediction performance suffers dramatically is such a situation.

Second, collaborative filtering algorithms can be used only when preference data for an item already exists in the database. In other words, these systems cannot handle queries pertaining to new items. For example, most collaborative filtering algorithms cannot help a user who needs to know whether a new movie being released is good. In such situations, the database has no information about the movie and the system is, therefore, unable to process such requests.

Third, the prediction algorithms in collaborative filtering are not based on a statistical model and therefore do not provide any measure of uncertainty in their predictions. While prediction uncertainty may not be crucial in low risk products such as movies or CD's, it can be important in other situations, such as the purchase of a stock or an expensive product. One way to deal with uncertainty in making recommendations is by explicitly accounting for risk aversion. Another, to which we limit our considerations in this paper, is to estimate the odds associated with alternative "payoffs," leaving the task of incorporating risk attitude up to the

[2]Another approach drawing on the field of information retrieval assesses similarity between people by treating each person as a vector of ratings and computing the cosine of the angle formed by two vectors. For details, see Salton and McGill (1983) and Breese et al.(1998).

judgement of the user.

Finally, most collaborative filtering systems do not incorporate attribute information explicitly. Typically, such systems are bootstrapped by creating "virtual users" representing particular tastes, such as a virtual "Action fan" who has high ratings for all Action movies and no other ratings. This implicitly, introduces product features (e.g., the Genre of a movie) in predicting preferences, but the implications of such indirect accounting of product features is not clear. Because collaborative filtering methods are purely correlational, they cannot provide any justification or explanation for a recommendation. An ability to explain why a particular item is being recommended may be important for building trust and for enhancing customer loyalty. To overcome the shortcomings of the collaborative filtering approach, we develop flexible yet simple statistical models that we describe in the next section.

3 An Overview of the Proposed Approach

For ease of exposition, we describe our approach in the context of the subsequent application, which recommends movies. The choice of the application is guided by three considerations. The first is simply the availability of a large commercial dataset. The second is that while a number of marketing researchers have modeled aggregate movie sales (e.g., Smith and Smith 1986; Dodds and Holbrook 1988; Sawhney and Eliashberg 1996; Eliashberg and Shugan 1997; Jedidi, et al. 1998), there is, to our knowledge, no related work on forecasting individual preferences. Third, movies are not only experience goods, but are also search goods (to the extent people prefer certain movie genres, and like or dislike certain actors and directors), and reputation goods (to the extent people pay attention to one or another movie critic). This makes it possible for us to combine all three types of information in our models.

We develop an ensemble of statistical methods, estimated using customer ratings on small, idiosyncratic subsets of products, to make customized recommendations over holdout items – in our case, new theater releases and video rentals (or re-releases) of older movies. We adopt a regression based approach and model the customer ratings in terms of product attributes, customer characteristics and expert evaluations. The models we develop differ in how they

account for unobserved sources of heterogeneity in customer preferences and product appeal structures. To the extent that parameter estimates reflect causal preference structures, they allow one to not only tell a person what he or she may like, but also why they may react in the predicted manner.

3.1 Customer Heterogeneity

The database consists of ratings provided by customers for many different movies. Let $i = 1$ to I represent customers and $j = 1$ to J represent movies. Customer i provides ratings for n_i movies in the database and let $M_i = \{j_1, j_2, \ldots, j_{n_i}\}$ be the index set of the n_i movies rated by the customer. Let r_{ij} represent the rating given by customer i for movie j where $j \in M_i$. Customers differ in the number of movies they rate, yielding an unbalanced data set. The total number of ratings in the database across all customers is given by $N = \sum_{i=1}^{I} n_i$. The observations for each customer can be used in specifying a customer level regression model as follows:

$$r_{ij} = \beta_{i1} + \beta_{i2}\text{Mvar}_j + e_{ij}, \quad e_{ij} \sim \mathcal{N}(0, \sigma^2). \tag{2}$$

$\forall j : j \in M_i$. In equation (2), Mvar_j represents a movie attribute[3] and the parameters β_{i1} and β_{i2} represent the preference structure of the customer.

If the database contains a large number of observations for each customer, then we can in principle estimate the above regression model for each customer. In many situations, however, the database is very sparse and only a few observations are available for some customers. We therefore cannot perform separate regressions for each customer.[4] We can, however, use a hierarchical Bayesian approach that adequately pools information across customers in order to make inferences pertaining to a specific customer. In this approach, a continuous mixture distribution is used to describe how the individual level parameters in Equation (2) vary across the customers in the population. For example, we can write the population model in terms of

[3]We initially describe our modeling approach using a single movie attribute and a single customer characteristic for ease of exposition. We later generalize our models to include all variables.

[4]Even if each customer in the database has enough observation for building an individual level regression model, the movies that he/she rated in the database may have a design matrix that does not allows estimation of all parameters. In such a case a separate model specification may be required for each customer thereby complicating the model building and prediction task.

a single customer characteristic as follows:

$$\begin{aligned} \beta_{i1} &= \mu_{11} + \mu_{12}\text{Demog}_i + \lambda_{i1} \\ \beta_{i2} &= \mu_{21} + \mu_{22}\text{Demog}_i + \lambda_{i2}, \end{aligned} \tag{3}$$

for $i = 1$ to I. The variable Demog_i represents a demographic variable for customer i and the random effects $\{\lambda_{i1}, \lambda_{i2}\}$, assumed to be distributed multivariate normal $\mathcal{N}(0, \mathbf{\Lambda})$, capture unobserved customer characteristics that may influence the customer preference structure. Equation (3) thus models the customer preference parameters in terms of both observed and unobserved customer characteristics. Substituting Equation (3) in Equation (2) yields the equation

$$r_{ij} = \mu_{11} + \mu_{12}\text{Demog}_i + \mu_{21}\text{Mvar}_j + \mu_{22}\text{Mvar}_j\text{Demog}_i + \lambda_{i1} + \lambda_{i2}\text{Mvar}_j + e_{ij}. \tag{4}$$

where, the $\mu's$ represent the fixed effects and document the influence of observed customer and movie variables and their interactions. The random effect λ_{i1} represents the unique aspects of person i regarding how this customer rates different movies and λ_{i2} is useful in modeling the interaction of unobserved customer characteristics for customer i and the observed movie attributes.

Generalizing Equation (4) to include all movie and customer variables, the customer heterogeneity model can be written as

$$\begin{aligned} r_{ij} &= \boldsymbol{x}_{ij}^{'}\boldsymbol{\mu} + \boldsymbol{w}_j^{'}\boldsymbol{\lambda}_i + e_{ij}, \\ e_{ij} &\sim \mathcal{N}(0, \sigma^2), \\ \boldsymbol{\lambda}_i &\sim \mathcal{N}(\mathbf{0}, \mathbf{\Lambda}), \end{aligned} \tag{5}$$

for $i = 1$ to I and $j \in M_i$. In the above equation, $\boldsymbol{x}_{ij}$ is a vector containing all observed movie attributes (i.e, genre variables and expert ratings) and person characteristics and their interactions and $\boldsymbol{w}_j$ is a vector that contains the observed movie attributes. The vector $\boldsymbol{\lambda}_i$ contains all the random effects pertaining to the ith customer. The covariance matrix $\mathbf{\Lambda}$ provides information about the extent of unobserved heterogeneity in customer preference structures. We now describe our second model that captures product heterogeneity.

3.2 Product Heterogeneity

Previous approaches to modeling heterogeneity in marketing have been applied to data involving a few products that are well described by observed attributes. For example, in conjoint analysis, a careful experimental design often yields a manageable number of product concepts. Moreover, the attribute levels constituting the profile provides a complete description of the product. Similarly, in consumer choice modeling, a limited number of choice alternatives are available and are well described by alternative specific constants and by attributes such as price and promotion. In such contexts, differences in customer preference structures primarily contribute to the heterogeneity in the data. In contrast, recommendation systems operate on databases that include ratings on a large number of products. Moreover, as in the case of movies and music, products cannot be described adequately in terms of a few observable attributes. Consumer preferences in such categories are shaped by myriad attributes that interact in intricate ways, leading to thematic differences that necessitate accounting for these complex yet unobserved product attributes (see Gershoff and West 1998). These unobserved movie attributes lead to differences in product appeal structures. Accommodating these differences among movies becomes crucial in modeling customer ratings. In this section we develop a model that accounts for unobserved movie attributes in modeling preferences.

Let $C_j = \{i_1, i_2, \ldots, i_{n_j}\}$ represent the index set of the n_j customers who rated movie j. Let r_{ji} represent the rating given by customer i for movie j where $i \in C_j$. Movies differ in the number of customers that provide ratings for them, yielding an unbalanced data set. The observations for movie j can be used in specifying a movie level regression model as follows:

$$r_{ji} = \boldsymbol{z}_i' \boldsymbol{\beta}_j + e_{ji}, \quad e_{ji} \sim \mathcal{N}(0, \sigma^2) \tag{6}$$

$\forall i : i \in C_j$, where $\boldsymbol{z}_i$ is a vector of customer characteristics for customer i and $\boldsymbol{\beta}_j$ is a vector of parameters for movie j that represents the movies appeal structure across customers. A hierarchical Bayesian model can be used to specify how the movies differ in their appeal structures across the population of movies. The population model that accounts for both observed and unobserved sources of heterogeneity can be written as

$$\boldsymbol{\beta}_j = \boldsymbol{w}_j' \boldsymbol{\mu} + \boldsymbol{\gamma}_j, \quad \boldsymbol{\gamma}_j \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Gamma}) \tag{7}$$

for $j = 1$ to J, where $\boldsymbol{w}_j$ contains the observed movie characteristics and $\boldsymbol{\gamma}_j$ represents the unobserved movie effects. The complete model can alternatively be written as

$$\begin{aligned} r_{ji} &= \boldsymbol{x}_{ji}^{'}\boldsymbol{\mu} + \boldsymbol{z}_i^{'}\boldsymbol{\gamma}_j + e_{ji}, \\ e_{ji} &\sim \mathcal{N}(0, \sigma^2), \\ \boldsymbol{\gamma}_j &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Gamma}), \end{aligned} \tag{8}$$

for $j = 1$ to J and $i \in C_j$. In the above equation $\boldsymbol{x}_{ji}$ contains all observed movie attributes and customer characteristics and their interactions. The variance matrix $\boldsymbol{\Gamma}$ provides information about the extent of unobserved heterogeneity in product appeal structures.

3.3 Customer and Product Heterogeneity

As is apparent from our earlier discussion, recommendation systems operate in contexts involving a large number of products and customers. It is therefore imperative to account for both customer and product heterogeneity in modeling preferences. we therefore combine both forms of heterogeneity. In the combined model, the rating r_{ij} for customer i can be written as follows:

$$\begin{aligned} r_{ij} &= \boldsymbol{x}_{ij}^{'}\boldsymbol{\mu} + \boldsymbol{z}_i^{'}\boldsymbol{\gamma}_j + \boldsymbol{w}_j^{'}\boldsymbol{\lambda}_i + e_{ij}, \\ e_{ij} &\sim \mathcal{N}(0, \sigma^2), \\ \boldsymbol{\lambda}_i &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Lambda}), \\ \boldsymbol{\gamma}_j &\sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Gamma}), \end{aligned} \tag{9}$$

$\forall j \in M_i$ and for all i. In the above model, $\boldsymbol{x}_{ij}$ is a vector of movie and customer variables, $\boldsymbol{z}_i$ contains the customer characteristics and $\boldsymbol{w}_j$ contains the movie variables. The random effects $\boldsymbol{\lambda}_i$ account for unobserved sources of customer heterogeneity and appear in the model interactively with the observed movie attributes. The random effects $\boldsymbol{\gamma}_j$ account for the unobserved source of heterogeneity in movie appeal structures and interact with the observed customer characteristics. Such a model provides a flexible framework for capturing differences in customer preference structures and movie appeal structures. We next describe estimation of parameters for our models.

4 Bayesian Inference

In this section, we describe our inference procedure for the model that includes both forms of heterogeneity. As is apparent from the previous section, the first two models are special cases of the third model as they can be obtained by restricting the appropriate variance parameters in the complete model. As an initial step, Bayesian inference requires specification of priors. We describe the prior distributions over the parameters in Appendix 1.

Inference in the Bayesian framework involves summarizing the joint posterior of all unknowns. As this posterior density is very complex, we use simulation based methods to obtain random draws from the posterior density. Inference can then be based on the empirical distribution of the draws. The complexity of the posterior density precludes the use of direct methods for obtaining these draws. We therefore use Markov Chain Monte Carlo (MCMC) methods. Specifically, our MCMC procedure involves Gibbs sampling (Gelfand and Smith 1990) steps to obtain the requisite draws. MCMC methods involve sampling parameter estimates from the full conditional distribution of blocks of parameters. In the context of our model, we need to generate random draws for the parameter blocks $\{\boldsymbol{\mu}, \{\boldsymbol{\lambda}_i\}, \{\boldsymbol{\gamma}_j\}, \sigma^2, \boldsymbol{\Gamma}, \boldsymbol{\Lambda}\}$. Each iteration of the MCMC sampler involves sequentially sampling from the full conditional distributions associated with each block of parameters. The full conditional distributions are given in Appendix 2.

The MCMC sampler is run for a large number of iterations. This iterative scheme of sequential draws generates a Markov chain that converges in distribution to the joint posterior under fairly general conditions (Tierney, 1994). After passing through an initial transient phase, the chain converges to the posterior distribution of parameters, and therefore the subsequent draws from the chain can be regarded as a sample from the posterior distribution. A large sample of draws can be obtained to approximate the posterior distribution to any degree of accuracy.

5 Movie Recommendations

In this section we describe an application of our modeling approach to movie recommendations on the Internet. We describe the data, the model specifications, the operationalization of the variables, and the null models. We then compare our the predictive ability of our models and discuss the results.

5.1 Datasets

The data were obtained from an actual recommendation system called EachMovie. This recommendation system, terminated in September 1997, was operated by DEC Systems research center for 18 months. The data consists of customer ratings for 1,628 different movies on a six point scale. The database also contains information pertaining to the genre of the movies and demographic information pertaining to customers. We used data from 340 movies and 2,000 customers for this application. For the 340 movies in our sample, we also collected expert evaluations from movie critics.

Our sample consists of 56,239 ratings. The average number of movies rated per customer is 29. The median number of movies is 19 with a minimum of 1 movie and a maximum of 235 movies per customer. The average number of ratings per movie is 163. The median number of ratings per movie, however, is 74 with a minimum of 1 rating and a maximum of 1,285 ratings. The data set is sparse, as each customer rated a small fraction of the population of movies and similarly, each movie is rated by a small fraction of the customers in the database. The overall sparseness of the database is evident from the fact that our 56,239 observations represent only 8% of the 680,000 ratings that are possible if each of the 2,000 customers in the sample rates each of the 340 movies. The data set is unbalanced as different customers rate different subsets of movies and different movies are rated by different subsets of people.

We divide our data into a calibration sample and four validation samples. The calibration sample contains 10,344 ratings on 228 movies and 986 customers. We construct the validation samples to reflect properties of customer interactions with the recommendation system. There

are four possible scenarios that can be described in terms of the information that is available on customers and movies in the database. First, a customer can either be an existing one, in which case customer preference data on this customer will be available in the database; or a new one, in which case only demographic information on this customer may be available. Second, a movie can be regarded as an "old" movie, if customer ratings for it are available in the database; or a "new" one, in which case, only genre and expert evaluations are available. Accordingly, we designed our four validation sets to reflect the four possible combinations of customer and movie types.

The first validation sample is generated by holding out 2,886 observations from the same set of customers and the same set of movies as in the calibration sample. We ensured that for each customer the movies in this validation sample are different from those the customer rated in the calibration sample. Similarly, for each movie, the customers who provide ratings for the movie in this validation sample are different from those in the calibration sample. We label this validation sample as the "OP/OM" sample ("Old Person-Old Movie").

The second validation sample contains ratings from the 986 customers in the calibration sample on the remaining 112 holdout movies in the dataset. We label this second validation sample as the "OP/NM" sample ("Old Person-New Movie"). The third validation sample contains ratings of the 1,014 holdout customers on the 228 movies in the calibration dataset. This reflects the situation in which a new customer interacts with the recommendation system. We label this third validation sample as the "NP/OM" sample ("New Person-Old Movie"). Finally, the last validation set contains observations pertaining to the 1,014 holdout customers on the 112 holdout movies. We label this fourth validation sample as the "NP/NM" sample ("New Person-New Movie"). The distribution of our samples is illustrated in Figure 1.

5.2 Model Specification and Variable Definition

We now describe the regression specification for the model that includes both forms of heterogeneity. The model can be written in terms of three components that include observed variables and two components that include unobserved random effects. The general template

for our model is as follows:

$$\begin{aligned} r_{ij} &= \text{Constant} + \text{Genre}_j + \text{Demographics}_i + \text{Expert Evaluations}_j \\ &\quad + \text{Customer Heterogeneity}_{ij} + \text{Movie Heterogeneity}_{ij} + e_{ij} \end{aligned} \tag{10}$$

where r_{ij} represents customer ratings on a six point scale from zero to five. We treat these ratings as interval scaled. We now describe each component of our model [5]

Genre: The genre variables are specified in terms of nine binary indicators that describe whether the movie pertains to one or more of the following genre categories: *Action, Art/Foreign, Classic, Comedy, Drama, Family, Horror, Romance* and *Thriller* . A movie can be simultaneously classified into more than one of the above genre categories. The genre effects are included in the model as follows:

$$\text{Genre}_j = \sum_k \mu_k \text{Genre}_{jk} \tag{11}$$

where Genre_{jk} refers to the genre variable k and μ's represent the fixed effects.

Demographics: The demographic variables include *Age* and *Sex* for the customers in the database and are represented in the model as follows:

$$\text{Demographics}_j = \sum_k \mu_k \text{Demographics}_{jk} \tag{12}$$

where Demographics_{jk} refers to the demographic variable k and μ's represent the fixed effects.

Expert Evaluations: The expert evaluations are from Roger Ebert (*Ebert*) of the Chicago Sun times, James Berardinelli (*James*) of ReelViews - an online film reviews site, the Videohound movie directory (*Bones*) and the Internet Movie Database (*IMDB*). The variables *Ebert, James* and *Bones* are on a nine point scale ranging from zero to four. The variable *IMDB* reflects the mean ratings of users on the Internet Movie Database with a range of zero to ten. The expert evaluations effects are included in the model as follows:

$$\text{Expert Evaluations}_j = \sum_l \mu_l \text{Expert}_{jl} \tag{13}$$

[5]Interactions between the observed variables were found to be non significant across all our specifications. We therefore ignore these for model parsimony.

where Expert_{jl} refers to the expert variable l and μ's represent the fixed effects as before.

Customer Heterogeneity: We model customer heterogeneity in two ways. First we allow for a customer specific random intercept. This captures any idiosyncrasies a customer may exhibit in rating movies in general. Second, we allow for customer specific interactions between the unobserved customer variables (random effects) and the observed descriptors (genre and expert evaluations) of a movie. The interactions allow us to model how different customers value different movie genre and the opinions of various experts. The customer heterogeneity effects are included in the model as follows:

$$\text{Customer Heterogeneity}_{ij} = \lambda_{i1} + \sum_k \lambda_{ik}\text{Genre}_{jk} + \sum_l \lambda_{il}\text{Expert}_{jl} \tag{14}$$

where Genre_{jk} refers to the genre variable k of movie j and Expert_{jl} refers to the expert evaluation l. The λ's represent the customer random effects and the $\boldsymbol{\lambda}_i$ vector is assumed to come from a multivariate normal population distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{\Lambda})$.

Product Heterogeneity: We model product heterogeneity similarly by allowing a) a movie specific random intercept which captures the movie equity and b) movie specific interactions between the unobserved movie attributes (random effects) and the observed customer demographics. The interactions allow us to model movie appeal structures, i.e., how different aspects of the movie appeal to different customer groups. The movie heterogeneity effects are included in the model as follows:

$$\text{Movie Heterogeneity}_{ij} = \gamma_{j1} + \sum_k \gamma_{jk}\text{Demographics}_{ik} \tag{15}$$

where Demographics_{ik} refers to the demographic variable k of customer i. The γ's represent the movie random effects and the $\boldsymbol{\gamma}_j$ vector is assumed to come from a multivariate normal population distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{\Gamma})$.

The summary statistics of the movie descriptors and the demographics are included in Table 1. The column "Mean 1" contains the mean across all 340 movies or across all 2,000 customers. For example, Table 1 indicates that 16.9% of the movies in our sample of 340 movies are classified as action movies; The mean rating of the movie critic Roger Ebert across all movies

in our sample is 2.745. The average age of the customers in our sample is 33 years, while 85% of the customers in our sample are male. The column "Mean 2" contains the mean across all 56,239 ratings in our sample. For example, Table 1 indicates that 28.4% of these ratings involved action movies, and 83.8% involved male customers.

5.3 Restricted Models

We compare our complete model with eleven other restricted models involving different combinations of the observed effects and heterogeneity specifications.

1. Our first class contains three restricted models that do not include any source of unobserved heterogeneity. Different specifications within this class are obtained depending on which categories of movie descriptors are included. The first model includes only genre variables, the second model includes only expert evaluations while the third includes both types of movie descriptors. All models include demographic variables.

2. Our second set consists of three restricted models that account only for customer heterogeneity. As in the first set of models, different specifications are obtained depending on which categories of movie descriptors are included. As there is no unobserved movie heterogeneity in this class of models, the impact of the demographic variables appears in the models through the fixed effects.

3. Our third class of three restricted models accounts only for the movie heterogeneity but allow for customer heterogeneity only through the demographic variables. Again, we distinguish between the three possible specifications that depend on the types of movie descriptors that are included. As unobserved sources of customer heterogeneity are not included, the movie descriptors appear in the model only through the fixed effects.

4. Our fourth class consists of two restricted models that capture both forms of heterogeneity. The first model includes only genre variables, while the second includes only expert evaluations.

This configuration of restricted models will allow us to understand the differential impacts of a) the two forms of heterogeneity, and b) the different types of movie descriptors in predicting customer preferences in the database. Using programs developed in the C language, we estimated a total of twelve models: the eleven restricted models and our complete model. As explained in Section 4, repeated draws were made from the sequence of full conditionals. The details of the priors for each of our models and aspects of the MCMC sequence are given in Appendix 1 and 2.

5.4 Results: Model Comparison

We use a) the marginal likelihood of the data (Kass and Raftery, 1995), and b) the DIC statistic (Spiegelhalter et. al. 1998) for model comparison. Both these criteria are based on the likelihood of a model and appropriately penalize a model for complexity. While the marginal likelihood has been traditionally used in the Bayesian literature, the DIC statistic has been recently suggested as an alternative criterion. Traditionally, marketing researchers have used penalized likelihood measures such as the AIC, BIC and CAIC for model comparison. These measures use the exact number of parameters in the model to impose a complexity penalty. In hierarchical models, however, the exact number of parameters is not known owing to the presence of random effects and therefore BIC and similar measures cannot be used. The DIC is simple to compute in most modeling contexts and can be written as

$$\text{DIC} = \bar{D}(\boldsymbol{\theta}) + p_D \tag{16}$$

where $D(\boldsymbol{\theta}) = -2LL(\boldsymbol{\theta})$ is the model deviance and is equal to twice the negative log likelihood. The vector $\boldsymbol{\theta}$ contains all parameters, including the random effects. The average deviance $\bar{D}$ is computed by taking the average of the deviance over the MCMC draws. The term p_D is used to penalize the model for complexity. It can be interpreted as the effective number of parameters in the model, and is computed as $p_D = \bar{D}(\boldsymbol{\theta}) - D(\bar{\boldsymbol{\theta}})$, where $D(\bar{\boldsymbol{\theta}})$ is the deviance calculated using the mean of the parameters $\bar{\boldsymbol{\theta}}$ obtained from the MCMC draws. The average deviance $\bar{D}$ is viewed as a measure of model fit. When comparing models, the model with the lowest DIC is considered the best model.

Table 2 reports the log-marginal likelihoods and the DIC statistics for all our models. It is clear from Table 2 that the complete model (last row) outperforms all other models on both model comparison criteria. The DIC statistics in the last row indicate that while this model is the most complex ($p_D = 1717$), it is also the best fitting (Fit= 31,488) and hence it outperforms the other models on the DIC statistic. In contrast, the first row of Table 2 shows that the model that does not capture any source of unobserved heterogeneity and includes only genre variables for describing the movies performs the worst.

A comparison of the different classes of models shows that the first set of models that do not allow for unobserved heterogeneity have the least support whereas, the models that include both sources of unobserved heterogeneity (the last set) have the greatest empirical support on both model comparison criteria.

A comparison of the second set of models with the third set shows that accounting for customer heterogeneity is more important than accounting for movie heterogeneity. The improvements in log-marginal likelihood are greater when customer heterogeneity is added (specially in the impact of expert variables) to the first set of models, than when movie heterogeneity is included as in the third set. The DIC statistics lead to the same conclusion that the customer heterogeneity models while more complex, are superior in fit and hence outperform the movie heterogeneity models.

5.5 Results: Parameters Estimates

Table 3 reports the posterior mean estimates for the fixed effects $\boldsymbol{\mu}$ and the standard deviations (i.e., the square root of the diagonal elements of $\boldsymbol{\Lambda}$ of $\boldsymbol{\Gamma}$) of the customer and movie random effects. Note that a variable can influence preferences either directly through the fixed effects or may influence the preferences by interacting with the random effects. The first row of Table 3 shows that the standard deviation of the customer-specific random effect associated with the intercept is 1.647. This implies that customers differ in their use of the rating scale. Similarly, the movie equity differs across the movies in the sample. This is evident from the fifth Column in Table 3 which shows that the standard deviation of the movie-specific random effect associated

with the intercept is 0.515. Most genre variables show insignificant fixed effects. On average, people like Action movies and Thrillers, and dislike Horror movies. The standard deviations of customer-specific random effects pertaining to the genre variables, however, are large for all genre variables and unambiguously indicate that customers differ in their preference structures. Thus, accounting for differences in the preference structures across customers is important for our application.

The fixed effects for James, Bones and IMDB are positive and significant and imply that the expert evaluations are in general positively associated with the ratings in the database. The random effects pertaining to the expert variables vary significantly across the users as is evident from the significant magnitudes of their standard deviation across customers. This implies that the association of the ratings with the expert evaluations varies across customers. Thus, accounting for expert evaluations is crucial in this application. Finally, the fixed effects for the Sex is insignificant whereas the coefficient for Age is significant. The demographics are an important component of the model as the standard deviations of the associated random effects across the movies reveal differences in movie appeal structures. Thus, it appears that movies appeal differentially to different demographic groups, but this differential appeal is based on unobserved movie attributes.

5.6 Results: Predictive Ability

We now compare the predictive ability of our models. Table 4 reports the root mean squared errors (RMSE) in prediction for all the models on the calibration data and on each of the four validation data sets. From Table 4, we see that in general, the RMSE statistics decrease in magnitude as we move from the top to the bottom of the table. Table 4 also shows that models that include customer heterogeneity have better fit than model that include movie heterogeneity alone. Comparing the RMSE statistics for the proposed model (last row) with those obtained from the restricted models, we see that the proposed model outperforms the other models on almost all the datasets.

While Table 4 focuses on RMSE statistics, ultimately, the test of our model is its facility in making accurate recommendations. We therefore transform the continuous predictions from our models to a zero to five scale so as to get an accurate picture of their predictive ability in terms of the actual ratings. To transform the predictions, we determine an optimal set of thresholds using the model predictions from the calibration data. We then use these optimal thresholds to classify the observations on the zero to five scale of the original ratings. We now consider the performance of our model on the four holdout datasets shown in Figure 1

The Traditional Collaborative Filtering Problem

We concentrate on the holdout predictions below. The predictions for the estimation set are, by and large, "better" than those for the holdout set. We first consider model recommendations on the old person-old movie validation data. Recall that we call a movie "old," if it is a released movie which at least some people have seen and for which ratings are available in our database. And we call the people in this dataset "old" because each of them has given a rating for at least one movie in the calibration sample. Predictions for such observations are possible using collaborative filtering. In fact this is the situation for which collaborative filtering is designed: it compares the similarity of preferences on a commonly-rated subset of movies, and predicts preferences for individual-level hold out sets of movies.

The "complete information" regarding the recommendation forecasts for this holdout data is shown in Table 5. The incidence matrix gives the raw frequencies, from which we obtain the column percentages reported in the lower half of the table. These percentages can be interpreted as follows: if we predict a rating of 0 for a movie not seen by a person, then the odds that a person will actually rate it a 0 is 65%, etc. That is, instead of making a point prediction, we can present to a user the odds that if they see a new movie, they will subsequently give it a certain rating. Three immediate observations can be made by examining Table 5. First, the diagonal numbers (perfect predictions) are not impressive. Second, the percentage of perfect matches decreases as one moves towards the center of the scale. Third, and most importantly, the greatest proportion of errors are nearest neighbors. Put another way, if we make a forecast of 5, then 86% of the true ratings are 4 or 5. These nearest neighbor percentages can be summarized

as follows.

Predicted	Nearest neighbors	Percent actual within ±1 point of prediction
5	4	86
4	3, 5	90
3	2, 4	77
2	1, 3	67
1	0, 2	66
0	1	76

It is clear from these numbers that if one were forced to make a point prediction, one would be most confident in making predictions within a rating point at the top end of the scale. This is a fortunate coincidence, which may or may not carry over to other applications. If asked for a recommendation, our model will of course never select movies with low ratings. And for movies with high predicted ratings (4 or 5), it will do quite well. In fact, if a recommendation systems were to be built using these results , one would likely restrict recommendations to those with ratings of 4 or 5. If the predicted rating is 4, the actual rating is no lower than 4 in 68% cases, and no lower than 3 in 90% of the cases: a "good surprise" 26% of the time, and a mild disappointment 22% of the time. Translated to raw numbers (see Table 5), this means that of the 1022 recommendations we make with a rating of 4, people who see these movies will, in 427 cases, come away with the exact same judgement about the movie as we predict; there will be 266 good surprises, and 224 will be mildly disappointed (we use the terms "happier" and "slightly" colloquially, for there is no saying how unhappy a slightly disappointed person might be). Among the remaining, 43 will probably not want to use the recommendation system again. Similarly, if people see the 325 recommendations for which we predict a rating of 5, then in 165 cases there will be complete agreement with our assessment, another 116 instances of mild disappointment, and 44 instances of reactions ranging from disappointment to sheer exasperation.

How disappointed will people be for not being told of a movie which they would have actually liked? The answer depends on our criteria for not recommending. If we never recommend movies for which we predict a rating of 0, then 0.5% (11 of 2886 people in our sample) people will not see a movie they would actually rate 4 or 5 were they to see it, because we failed to

tell them about it. The 2 and 3 predictions are murky grounds. There is much that is good to miss, and much that one might want to see. To summarize, if one only looks at exact matches, our prediction model is far from perfect. But if one thinks of how it might be used in making recommendations, its not so bad at all.

At this point one might reasonably ask how we compare with actual recommendation systems on the market. Unfortunately, commercial vendors are loath to share proprietary implementations, and any implementation of algorithms we execute is open to criticism by these companies. Fortunately, a recent report (Breese et. al., 1998), written by researchers at *Microsoft, Firefly's* parent company, gives us some bases for comparing our method with implementations of four collaborative filtering algorithms for the same database of movies used in the present paper. These four methods are (i) collaborative filtering, (ii) Bayesian networks, (iii) Bayesian clustering (mixture/latent class models), and (iv) vector similarity. As is common in the literature on collaborative filtering, Breese et. al. use

$$\text{MAD} = \frac{\sum |\text{predicted rating} - \text{true rating}|}{\text{number of observations}},$$

to compare the performances of these methods. The results of their analysis can be summarized as follows: on a 0-5 rating scale, the lowest mean absolute deviations between actual and predicted ratings is 0.994. The lowest MAD was obtained by using all but one holdout movie in the estimation, and is for the simplest (correlational) collaborative-filtering model, which outperforms a mixture (latent class) model, a Bayesian network model, and a vector similarity model.

The best predictions (lowest MADs) from their analysis of these four models (keeping one movie per person for prediction) are compared to "split-half" predictions from (i) an aggregate regression model that makes average (i.e., not customized) recommendations and (ii) our model.[6]

[6]We caution that a direct comparison of results is not possible as the calibration data set we use differs from the exact data set used in their study

Model	Method	No. of movies per person for estimation	No. of movies per person for prediction	MAD
Collaborative Filtering Bayesian Clustering Bayesian Network Vector Similarity	Keep 1 movie/person for prediction. Use all others for estimation.	mean=46.5 median=26	mean=1 median=1	0.994 1.103 1.006 2.136
Regression Ours	Random split for estimation/prediction.	mean=12.82 median=8	mean=5.33 median=3	1.094 0.905

Observe that the MAD values of around 1 on a 5 point scale are not exactly impressive for any model. Surprisingly, not only does the proposed model outperform the others, the aggregate regression – an *uncustomized* recommendation system – is comparable to the more complex methods (Bayesian Clustering, Bayesian Networks) reported in the paper.

Why does regression do so well? Not because it is a better model, but because MAD is not very informative about how a model fails. To see this, compare the row/column marginals of the incidence matrices in Tables 5 and 6. Note that (i) there are 54.19% true 3/4 ratings, (ii) our model predicts 3/4 ratings in 66.28% cases, and (iii) regression predicts 3/4 in just 90.3% cases. So in an overwhelming number of cases, a recommendation based on regression will tell a person he/she will give a 3/4 rating after seeing a movie.

While it should certainly be useful to compare these alternatives recommendation systems in terms of the implications one can draw from the "complete information" shown in Table 5 for our model, we have no access to their proprietary programs, but urge them to provide similar validation results.

We now consider three other prediction situations in which collaborative filtering can make no recommendation at all.

Old People, New Movies: A new movie is released. No one (at least in the in the sample of users) has seen the movie, although some experts have, and one obviously knows movie characteristics.

New People, Old Movies: A new person registers to use the recommendation system, providing some demographic information. Upon entering the site, the person asks for recommendations *without rating any movies at all.*

New People, New Movies: This is the same situation as above, except that the person wants recommendations about new movie releases (i.e., movies for which no person in the sample has given ratings).

A priori, the quality of recommendations ought to decrease as the available information on a person or movie decreases. In terms of MAD, this is indeed the case:

Person	Movie	MAD
Old	Old	0.905
Old	New	0.971
New	Old	1.008
New	New	1.122

Nevertheless, even in the worst case above, the deterioration in the MAD values is not terrible, for it changes from about 1 point for all the methods above to 1.12 in the worst case (new person, new movie). But as noted above, one needs to examine the full distribution of predictions to assess how well the method does in each case. Tables 7 to 9 give the complete distributions. The odds of "true" rating when one predicts ratings of 5 (which correspond to movies one might recommend), are summarized below.

		True rating					
Person	Movie	5	4	3	2	1	0
Old	Old	0.5077	0.3569	0.00923	0.0308	0.0031	0.0092
Old	New	0.4926	0.3202	0.0927	0.0464	0.0254	0.0227
New	Old	0.5114	0.3186	0.1186	0.0240	0.0108	0.0168
New	New	0.4279	0.2443	0.1377	0.0721	0.0393	0.0787

The above distributions do not appear to differ markedly from each other, except for "new person" and "new movie." But note that for "old person-old movies," the percentage of movies predicted to have 5 ratings decreases from 11.26% to 7.17% while the percentage of movies with

"true" ratings of 5 drops from 19.26% to 17.14%. Put another way, the model becomes more conservative in predicting a 5 rating as one goes from old to new movies. This conservative pattern of predictions become even more pronounced as one uses lesser information to make predictions. Thus, in the worst case (new person-new movie), the model predicts ratings of 5 in 3.95% cases, while there are 17.76% movies with a true rating of 5, a number close to the 3.12% for the aggregate regression in Table 6. A similar conservative pattern of predictions is observed at the other end of the scale (i.e., for 0 and 1 ratings).

Now consider the following odds for a person's "true" rating given a prediction of 4.

		True rating					
Person	Movie	5	4	3	2	1	0
Old	Old	0.2330	0.5300	0.01880	0.0330	0.0100	0.0060
Old	New	0.2528	0.4208	0.2121	0.0624	0.0262	0.0258
New	Old	0.2897	0.3751	0.1965	0.0650	0.0246	0.0492
New	New	0.2489	0.3518	0.2181	0.0800	0.0383	0.0629

Here one sees a more noticeable decline in the performance of the model, with a significant increase in the number of true 3 ratings as less information becomes available. Still, if one considers the odds of a 4 prediction being a true 4 or 5 rating, the 0.6 odds in the worst case (new movie, new person) are not too bad when compared to the odds of 0.763 in the best case (old movie, old person).

To summarize, as far as comparison with collaborative filtering is concerned, the proposed model is substantially better for two reasons. First, it does better in those situations when both collaborative filtering and our method can be used to make recommendations (i.e., old person-old movie). Second, it can make recommendations in those situations where collaborative filtering cannot be used. As one might expect, the lesser the information one has on a person or a movie, the less accurate the predictions.

As long as one is interested in making recommendations, the present model performs quite well, because one is concerned with making accurate prediction of movies with high (4 or 5) ratings, provided there are enough movies for which these ratings are predicted. But if one is interested in responding to queries (should I see this movie or not), then even in the best

case (old movie, old person), a predicted rating of 2 or 3 has unacceptably high odds of being some other true rating. This is an area in which model improvement is especially desirable. That said, we believe that it is better for any recommendation system to not just make a point prediction but to predict odds for true ratings in the manner discussed here, together with the count (number of observations) on which the odds are based. This, in the end, is much better for making informed choices than a forced prediction, which in situations involving risky decisions, can lead to bad decisions by a person who place undue trust in point predictions.

6 Summary and Conclusions

Recommendation systems are being increasingly used for mass customization of information products on the internet. In this paper we described an attribute based approach for making customized product recommendations based on a database of preferences from customers. Our modeling approach accounts for differences in customer preference structures over product attributes and simultaneously captures the variation in product appeal structures. We estimated our models on a database of customer preferences for movies. Our results show that the model that incorporates multiple sources of information, i.e., movie genre variables, evaluations from experts, and the demographic characteristics of customers and accounts for unobserved customer and movie heterogeneity outperforms restricted models in terms of fit and predictive ability.

Our models provide significant practical advantages over collaborative filtering algorithms that are used in commercial implementations of recommendation systems. Our models can make recommendations in many different scenarios that cannot be handled by collaborative filtering approaches. For example, collaborative filtering methods can only make recommendations of products for which preference ratings are already available in the database. In addition, as collaborative filtering methods are not based on a statistical model of preferences, they cannot account for prediction uncertainty. Moreover, collaborating filtering approaches cannot provide any explanation or justification for a recommendation. The previous two shortcomings of conventional collaborative filtering methods will limit their applicability in risky situations.

The simple yet flexible models we developed in this paper can be generalized in many directions. We developed models for situations in which explicit ratings are available in the database. Clearly, in many situations, only implicit information regarding customer preferences may be available. For example, instead of stated preference data, only revealed preference data gleaned from previous encounters may be available. Our methods can be generalized to handle such data. We used a linear model to describe preferences. The literature on neural networks and non-parametric methods suggest that prediction performance can be improved by incorporating nonlinearity in models. Future research can investigate, for instance, how using radial basis function neural networks or multilayer perceptrons may impact the quality of the recommendations. We used continuous mixture distributions to characterize heterogeneity. Future research can explore finite mixture models to investigate how the predictive ability of such models compares with ours. While we focused on movie predictions, researchers and marketers can apply our models to investigate recommendations in other domains. For example, generalizations of our model can be used in retail contexts for cross-selling products to customers.

We focused on one type of software agent, e.g., recommendation systems. Software agents of many different guises abound on the Internet. Clearly there is a need to study how other types of information agents can be gainfully used by marketers. For example, negotiation agents, matchmaking agents, and agents designed to participate in auctions are directly relevant for marketers. The approaches and methodologies that have evolved in the marketing literature to understand customer preferences and other aspects of consumer behavior can be used to enrich the emerging literature on software agents. In addition, marketing researchers can draw upon the economics and computer science literatures on bargaining and matching-markets to study negotiation and matchmaking agents. The new applications of information agents will also require advances in data collection and analysis procedures; marketing researchers are eminently poised to contribute significantly in these areas.

References

Allenby, G., and J. Ginter (1995), "Using Extremes to Design Products and Segment Markets," *Journal of Marketing Research*, 32, 392-403.

Allenby, G.M., and P.E. Rossi (1999), "Marketing Models of Consumer Heterogeneity," in Special Issue : Marketing and Econometrics: (eds. Wansbeek, T and Wedel M.) *Journal of Econometrics*, Vol 89 1-2 pp. 57-78.

Breese, John, S., David Heckerman and Carl Kadie (1998), "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," technical report MSR-TR-98-12, Microsoft Research, October.

Balabanović, Marko and Yoav Shoham (1997), "Fab: Content Based Collaborative Recommendation," *Communications of the Association for Computing Machinery,* 40 (3), 66-72.

Dodds, John C. and Morris B. Holbrook (1988), "What's an Oscar Worth? An Empirical Estimation of the Effect of Nominations and Awards on Movie Distribution and Revenues,"in *Current Research in Film: Audiences, Economics and the Law,* Vol 4, B. A. Austin, ed., Norwood, NJ: Ablex Publishing Co.

Eliashberg, Jehoshua and Steven M. Shugan (1997), "Film Critics: Influencers or Predictors?" *Journal of Marketing*, 61, 2, 68-78

Foner, Leonard N. (1993), "What is an Agent, Anyway? A Sociological Case Study" working paper, MIT Media Labs.

Gelfand, A. E., and A.F.M. Smith (1990), "Sampling-Based Approaches to Calculating Marginal Densities," *Journal of the American Statistical Association*, **85**, 972-985.

Gershoff, Andrew and Patricia M. West (1998), "Using a Community of Knowledge to Build Intelligent Agents," *Marketing Letters*, 9 79-91.

Goldberg, David, David Nichols, Brian M. Oki and Douglas Terry (1992), "Using Collaborative Filtering to Weave an Information Tapestry," *Communications of the Association for Computing Machinery,* 35 (12), 61-70.

Jedidi, Kamel, R. E. Krider and C B. Weinberg (1998), "Clustering at the Movies," *Marketing Letters*, 9(4), 393-405.

Kass, R. E., and Raftery, A. E., (1995), "Bayes Factors," *Journal of American Statistical Association*, 90, 773-795.

Kay, Alan (1984), "Computer Software," *Scientific American,* 251(3) September, 53-59.

Kay, Alan (1990), "User Interface: A Personal View," in *The Art of Human-Computer Interface Design,* ed: Brenda Laurel, MA: Addison-Wesley, 191-207.

Levy, Steven (1993), "Let Your Agents Do the Walking," *MacWorld,* May, p. 42.

Maes, Pattie (1994), "Agents that Reduce Work and Information Overload," *Communications*

of the Association for Computing Machinery, 37(7) July, 31-40.

Negroponte, Nicholas (1970), "*The Architecture Machine,* Boston: MIT Press.

Rossi, P. R. McCulloch, and G. Allenby (1996), "On the Value of Household Purchase History Information in Target Marketing", *Marketing Science,* 15, 321-340.

Salton, Gerard and Christopher Buckley (1988), "Term weighting Approaches in Automatic Text Retrieval," *Information Processing and Management,* 24 (5), 513-23.

Salton, Gerard and M. McGill (1983), *Introduction to Modern Information Retrieval,* New York: McGraw Hill.

Sarwar, Badrul M., Joseph A. Konstan, Al Borchers, John Herlocker, Brad Miller and John Riedl (1998), "Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System," in *Proceedings of the 1998 Conference on Computer Suppoted Collaborative Work.*

Sawhnew, Mohanbir S. and Jehoshua Eliashberg (1996), "A Parsimonious Model for Forecasting Gross Box Office Revenues of Motion Pictures," *Marketing Science,* 15 (2), 113-131.

Shardanand, Upendra and Pattie Maes(1995), "Social Information Filtering: Algorithms for Automating," in *Proceedings of the ACM CHI'95 Conference on Human Factors in Computing Systems," 210-17.*

Smith, Sharon, P. and V. Kerry Smith (1986), "Successful Movies: A Preliminary Empirical Analysis," *Applied Economics,* 18, 501-507.

Spiegelhalter, David, J., N.G. Best, and B. P. Carlin (1998) "Bayesian Deviance, the Effective Number of Parameters, and the Comparison of Arbitrarily Complex Models", Technical Report, Division of Biostatistics, University of Minnesota.

Tierney, L. (1994), "Markov chains for exploring posterior distributions (with discussion." *Annals of Statistics,* 22, 1701-1762.

West, P. M., D. Ariely, S. Bellman, E. Bradlow, J. Huber, E. Johnson, B. Kahn, J. Little and D. Schkade (1999), "Agents to the Rescue?" *Marketing Letters,* special issue, July, 207-40.

Appendix 1: Prior Distributions

The unknown parameters for the model are $\beta = \{\mu, \Lambda, \Gamma, \sigma\}$. In this paper, we specify the prior distribution over β as a product of independent priors,

$$p(\beta) \;=\; p(\mu)\, p(\Lambda)\, p(\Gamma)\, p(\sigma) \qquad \text{(i)}$$

We use proper, but diffuse priors over all model parameters. The priors can be written as follows:

(a) The prior for the fixed effects μ can be chosen to be multivariate normal $\mathcal{N}(\eta, C)$. The covariance matrix C can be assumed to be diagonal with its variance elements set to large values to reflect lack of knowledge about μ. When the prior is diffuse, the exact value for η does not matter and this can therefore be set to zero. We therefore use $\eta = \mathbf{0}$ and $C = 1000I$, where I is the identity matrix.

(b) The precision matrix Λ^{-1} associated with the population distribution, $\lambda_i \sim \mathcal{N}(\mathbf{0}, \Lambda)$, is a $(m+1) \times (m+1)$ positive definite matrix (where m is the number of movie descriptors). The precision matrix Γ^{-1} associated with the population distribution $\gamma_j \sim \mathcal{N}(\mathbf{0}, \Gamma)$, is a $(p+1)\times(p+1)$ positive definite matrix (where p is the number of customer characteristics).

In keeping with standard Bayesian analysis of linear models, we assume Wishart priors: $\mathcal{W}(\imath, (\imath L)^{-1})$ for the precision matrix Λ^{-1}, and $\mathcal{W}(\jmath, (\jmath G)^{-1})$ for the precision matrix Γ^{-1}. The matrices L and G can be considered as the expected prior variances of the λ_i's and γ_j's, respectively. Smaller values for $\imath$ and $\jmath$ correspond to more diffuse prior distributions. We set $\imath = 16$, $\jmath = 5$, $G = diag(0.001)$ and $L = diag(0.001)$.

(c) The prior for the error variance σ^2 is chosen to be inverse gamma $\mathcal{IG}(a, b)$. We set $a = 3$ and $b = 1000$.

Appendix 2: Full Conditional Distributions for the Full Model

(a) The parameter μ can be generated from the multivariate normal full conditional distribution given by

$$p(\mu \mid \{r_{ij}\}, \{\lambda_i\}, \{\gamma_j\}, \sigma^2) \sim \mathcal{N}(\hat{\mu}, V_\mu) \qquad A.1$$

where $V_\mu^{-1} = C^{-1} + \sigma^{-2} X' X$, and $\hat{\mu} = V_\mu(\sigma^{-2} X' \tilde{r} + C^{-1}\eta)$. The matrix X is obtained by stacking row by row all the row vectors x_{ij}'. The vector $\tilde{r}$ is obtained by stacking all the elements $\tilde{r}_{ij} = r_{ij} - z_i'\gamma_j - w_j'\lambda_i$, for all the person-movie pairs.

(b) The full conditional distribution of the error variance σ^2 is inverse gamma, and is given by

$$p(\sigma^2 \mid \{r_{ij}\}, \{\lambda_i\}, \{\gamma_j\}, \mu) \sim \mathcal{IG}\left(\frac{N}{2} + a\left[\frac{1}{2}(\tilde{r} - X\mu)'(\tilde{r} - X\mu) + b^{-1}\right]^{-1}\right) \qquad A.2$$

(c) The customer random effects λ_i can be generated from the multivariate normal full conditional distribution given by

$$p(\lambda_i \mid \{r_{ij}\}, \mu, \{\gamma_j\}, \sigma^2, \Lambda) \sim \mathcal{N}(\hat{\lambda}_i, V_i) \qquad A.3$$

where $V_i^{-1} = \Lambda^{-1} + \sigma^{-2} W_i' W_i$, and $\hat{\lambda}_i = V_i(\sigma^{-2} W_i' \tilde{r}_i)$. The matrix W_i is obtained by stacking row by row all the row vectors w_j' for j belonging to the index set of customer i's movies, M_i, . The vector $\tilde{r}_i$ is obtained by stacking the elements $\tilde{r}_{ij\lambda} = r_{ij} - x_{ij}'\mu - z_i'\gamma_j$, for all the movies $j \in M_i$ of customer i.

(d) The movie random effects γ_j can be generated from the multivariate normal full conditional distribution given by

$$p(\gamma_j \mid \{r_{ij}\}, \mu, \{\lambda_i\}, \sigma^2, \Gamma) \sim \mathcal{N}(\hat{\gamma}_j, V_j) \qquad A.4$$

where $V_j^{-1} = \Gamma^{-1} + \sigma^{-2} Z_j' Z_j$, and $\hat{\gamma}_j = V_j(\sigma^{-2} Z_j' \tilde{r}_j)$. The matrix Z_j is obtained by stacking all the row vectors z_i' for i in C_i, the index set of movie j's customers. The vector $\tilde{r}_j$ is obtained by stacking the elements $\tilde{r}_{ij\gamma} = r_{ij} - x_{ij}'\mu - w_j'\lambda_i$ for all the customers $i \in C_j$ of movie j.

(e) The full conditional distribution of the precision matrix $\mathbf{\Lambda}^{-1}$ of the unobserved customer characteristics is Wishart, and is given by

$$p(\mathbf{\Lambda}^{-1} \mid \{\boldsymbol{\lambda}_i\}) \sim \mathcal{W}\left(\left[\sum_{i=1}^{I} \boldsymbol{\lambda}_i \boldsymbol{\lambda}_i' + \imath \boldsymbol{L}\right]^{-1}, \ \imath + I\right) \qquad A.5$$

(f) The full conditional distribution of the precision matrix $\mathbf{\Gamma}^{-1}$ of the unobserved movie characteristics is also Wishart, and is given by

$$p(\mathbf{\Gamma}^{-1} \mid \{\boldsymbol{\gamma}_j\}) \sim \mathcal{W}\left(\left[\sum_{j=1}^{J} \boldsymbol{\gamma}_j \boldsymbol{\gamma}_j' + \jmath \boldsymbol{G}\right]^{-1}, \ \jmath + J\right) \qquad A.6$$

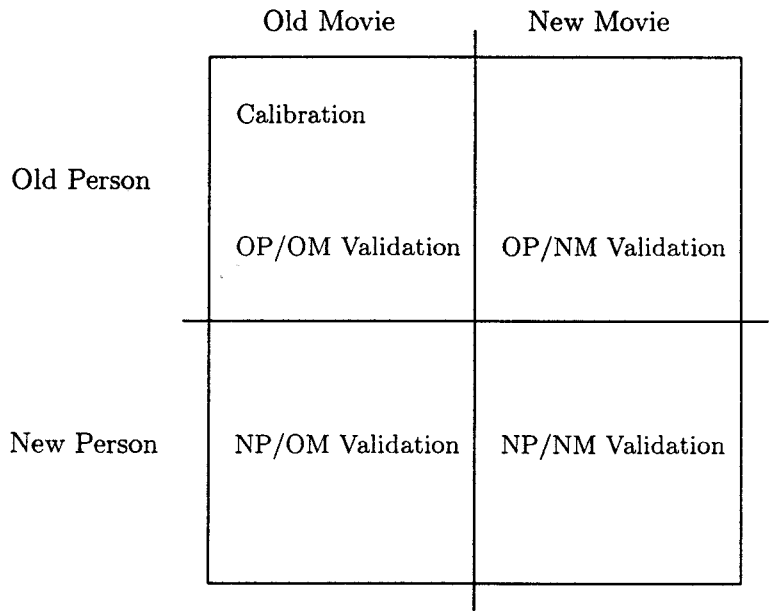

Figure 1: Calibration and Validation Samples

Table 1: Summary Statistics

	Variables	Mean 1 [1]	(Std 1)	Mean 2 [2]	(Std 2)
Genre Variables	Action	0.169	(0.375)	0.284	(0.451)
	Art/Foreign	0.143	(0.350)	0.080	(0.272)
	Classic	0.003	(0.054)	0.007	(0.085)
	Comedy	0.300	(0.459)	0.307	(0.461)
	Drama	0.426	(0.495)	0.384	(0.486)
	Family	0.079	(0.270)	0.108	(0.310)
	Horror	0.044	(0.205)	0.074	(0.262)
	Romance	0.146	(0.353)	0.152	(0.359)
	Thriller	0.160	(0.367)	0.179	(0.384)
Expert Variables	Ebert	2.745	(0.827)	2.888	(0.831)
	James	2.653	(0.761)	2.738	(0.779)
	Bones	2.567	(0.651)	2.737	(0.656)
	IMDB	7.072	(1.239)	6.974	(1.274)
Demographic Variables	Sex	0.851	(0.356)	0.838	(0.369)
	Age	33.01	(11.28)	31.63	(10.63)

1- Mean 1 and Std 1 are the mean and std. dev. across all 340 movies or all 2,000 customers.

2- Mean 2 and Std 2 are the mean and std. dev. across all 56,239 ratings.

Table 2: Model Comparison Statistics

Models		Log-Marginal Likelihood	DIC Statistics		
Heterogeneity	Movie Attributes		Fit D	Complexity p_D	DIC
No Heterogeneity	Genre Only	-18,801	37,589	13	37,602
	Expert Only	-18,398	36,788	8	36,796
	Genre & Expert	-18,327	36,638	17	36,655
Customer Heterogeneity	Genre Only	-17,581	34,135	1020	35,155
	Expert Only	-17,162	33,429	900	34,329
	Genre & Expert	-16,909	32,215	1,501	33,716
Movie Heterogeneity	Genre Only	-18,072	35,825	275	36,100
	Expert Only	-18,067	35,834	259	36,093
	Genre & Expert	-18,066	35,830	260	36,090
Movie & Customer Heterogeneity	Genre Only	-16,793	32,118	1,390	33,508
	Expert Only	-16,840	32,502	1,146	33,648
	Genre & Expert	-16,675	31,488	1,717	33,205

All models include demographic variables.

Table 3: Parameter Estimates for the Complete Model

	Variables	Fixed Effects μ	Std. Dev. across Customers	Std. Dev. across Movies
	Intercept	0.325 (0.371)*	1.647	0.515
Genre Variables	Action	0.341 (0.131)	0.407	
	Art/Foreign	0.053 (0.148)	0.573	
	Classic	-0.085 (0.571)	0.318	
	Comedy	0.210 (0.127)	0.300	
	Drama	0.124 (0.117)	0.257	
	Family	0.048 (0.168)	0.321	
	Horror	-0.411 (0.226)	0.347	
	Romance	-0.084 (0.137)	0.301	
	Thriller	0.319 (0.162)	0.292	
Expert Variables	Ebert	0.085 (0.062)	0.127	
	James	0.231 (0.089)	0.210	
	Bones	0.266 (0.088)	0.229	
	IMDB	0.125 (0.048)	0.087	
Demographic Variables	Sex	0.018 (0.074)		0.155
	Age	0.006 (0.003)		0.024
	σ^2	1.229 (0.021)		

* Standard deviations across MCMC iterations are in parentheses.

Table 4: Root Mean Squared Errors

		Calibration Sample	Old Person Old Movie	Old Person New Movie	New Person Old Movie	New Person New Movie
No Heterogeneity	Genre Only	1.488	1.504	1.544	1.421	1.500
	Expert Only	1.431	1.453	1.460	1.376	1.444
	Genre & Expert	1.420	1.442	1.476	1.370	1.458
Movie Heterogeneity	Genre Only	1.349	1.442	1.542	1.349	1.468
	Expert Only	1.350	1.417	1.473	1.348	1.460
	Genre & Expert	1.350	1.418	1.488	1.349	1.478
Customer Heterogeneity	Genre Only	1.196	1.313	1.307	1.414	1.516
	Expert Only	1.163	1.278	1.299	1.369	1.438
	Genre & Expert	1.061	1.241	1.306	1.361	1.456
Movie & Person Heterogeneity	Genre Only	1.063	1.233	1.367	1.339	1.483
	Expert Only	1.192	1.287	1.296	1.383	1.432
	Genre & Expert	1.012	1.216	1.295	1.337	1.446

Table 5: Proposed Model - Validation Sample OP/OM

Incidence Matrix

Actual	Predicted 0	1	2	3	4	5	Total	%
0	130	67	56	58	24	3	338	11.71
1	23	21	26	55	19	1	145	5.02
2	19	31	44	123	62	10	289	10.01
3	18	42	82	296	224	30	692	23.98
4	7	13	43	266	427	116	872	30.21
5	4	6	16	93	266	165	550	19.26
Total	201	180	267	891	1022	325	2886	
%	6.96	6.24	9.25	30.87	35.41	11.26		100

Column Percentages

Actual	Predicted 0	1	2	3	4	5
0	64.68	37.22	20.97	6.51	2.35	0.92
1	11.44	11.67	9.74	6.17	1.86	0.31
2	9.45	17.22	16.48	13.80	6.07	3.08
3	8.96	23.33	30.71	33.22	21.92	9.23
4	3.48	7.22	16.10	29.85	41.78	35.69
5	1.99	3.3	5.99	10.44	26.03	50.77

Table 6: Regression - Validation Sample OP/OM

Incidence Matrix

Actual	Predicted 0	1	2	3	4	5	Total	%
0	0	37	29	147	124	1	338	11.71
1	1	13	11	68	49	3	145	5.02
2	1	15	12	131	126	4	289	10.01
3	0	17	28	288	346	13	692	23.98
4	0	13	9	247	568	35	872	30.21
5	0	1	3	106	406	34	550	19.26
Total	2	96	92	987	1619	90	2886	
%	0.07	3.33	3.19	34.20	56.10	3.12		100

Column Percentages

Actual	Predicted 0	1	2	3	4	5
0	76	37.9	17.7	4.9	0.6	0
1	16	25.5	17.6	7.0	1.0	0.1
2	5.3	18.3	23.5	14.6	3.3	0.2
3	1.8	14.0	29.0	36.9	18.8	1.6
4	0.9	4.0	10.5	31.5	53.0	27.8
5	0	0.3	1.6	5.2	23.3	70.2
Total	1.7	2.8	5.7	23.1	52.0	100

Table 7: Proposed Model - Validation Sample OP/NM

Incidence Matrix

Actual	Predicted						Total	%
	0	1	2	3	4	5		
0	549	273	324	467	136	26	1775	11.14
1	84	113	177	417	138	29	958	6.01
2	91	160	298	806	329	53	1737	10.90
3	84	155	455	1912	1119	106	3831	24.04
4	56	95	271	1895	2220	366	4903	30.77
5	20	35	93	686	1334	563	2731	17.14
Total	884	831	1618	6183	5276	1143	15935	
%	5.55	5.21	10.15	38.80	33.11	7.17		100

Column Percentages

Actual	Predicted					
	0	1	2	3	4	5
0	62.10	32.85	20.02	7.55	2.58	2.27
1	9.50	13.60	10.94	6.74	2.62	2.54
2	10.29	19.25	18.42	13.04	6.24	4.64
3	9.50	18.65	28.12	30.92	21.21	9.27
4	6.33	11.43	16.75	30.65	42.08	32.02
5	2.26	4.21	5.75	11.09	25.28	49.26

Table 8: Proposed Model - Validation Sample NP/OM

Incidence Matrix

Actual	Predicted 0	1	2	3	4	5.	Total	%
0	6	132	158	365	274	14	949	8.09
1	2	31	95	254	137	9	528	4.50
2	1	55	126	510	362	20	1074	9.16
3	1	69	191	1108	1095	99	2563	21.86
4	0	54	165	1364	2090	266	3939	33.59
5	1	16	52	564	1614	427	2674	22.80

Column Percentages

Actual	Predicted 0	1	2	3	4	5
0	54.55	36.97	20.08	8.76	4.92	1.68
1	18.18	8.68	12.07	6.10	2.46	1.08
2	9.09	15.41	16.01	12.24	6.50	2.40
3	9.09	19.33	24.27	26.60	19.65	11.86
4	0.00	15.13	20.97	32.75	37.51	31.86
5	9.09	4.48	6.61	13.54	28.97	51.14