http://pubsonline.informs.org/journal/mksc/

MARKETING SCIENCE
Vol. 37, No. 6, November–December 2018, pp. 987–1008
ISSN 0732-2399 (print), ISSN 1526-548X (online)

Probabilistic Topic Model for Hybrid Recommender Systems: A Stochastic Variational Bayesian Approach

Asim Ansari,[a] Yang Li,[b] Jonathan Z. Zhang[c]

[a] Marketing Division, Columbia Business School, Columbia University, New York, New York 10027; [b] Marketing, Cheung Kong Graduate School of Business, Beijing 100738, China; [c] Department of Marketing and International Business, Foster School of Business, University of Washington, Seattle, Washington 98195

Contact: maa48@columbia.edu, http://orcid.org/0000-0001-6964-6297 (AA); yangli@ckgsb.edu.cn, http://orcid.org/0000-0002-7380-1883 (YL); zaozao@uw.edu, http://orcid.org/0000-0002-9559-0626 (JZZ)

Received: September 16, 2016
Revised: September 21, 2017; March 15, 2018
Accepted: April 19, 2018
Published Online in Articles in Advance: November 21, 2018

https://doi.org/10.1287/mksc.2018.1113

Abstract. Internet recommender systems are popular in contexts that include heterogeneous consumers and numerous products. In such contexts, product features that adequately describe all the products are often not readily available. Content-based systems therefore rely on user-generated content such as product reviews or textual product tags to make recommendations. In this paper, we develop a novel covariate-guided, heterogeneous supervised topic model that uses product covariates, user ratings, and product tags to succinctly characterize products in terms of latent topics and specifies consumer preferences via these topics. Recommendation contexts also generate big-data problems stemming from data volume, variety, and veracity, as in our setting, which includes massive textual and numerical data. We therefore develop a novel stochastic variational Bayesian framework to achieve fast, scalable, and accurate estimation in such big-data settings and apply it to a MovieLens data set of movie ratings and semantic tags. We show that our model yields interesting insights about movie preferences and makes much better predictions than a benchmark model that uses only product covariates. We show how our model can be used to target recommendations to particular users and illustrate its use in generating personalized search rankings of relevant products.

History: Peter Rossi served as the senior editor and Michel Wedel served as associate editor for this article.
Supplemental Material: Data are available at https://doi.org/10.1287/mksc.2018.1113.

Keywords: hybrid recommendation models • personalized search • user-generated content • probabilistic topic models • big data • scalable inference • stochastic variational Bayes

1. Introduction

Over the last decade, e-commerce firms and digital content providers, such as Amazon, Netflix, and the *New York Times*, have become increasingly reliant on recommender systems to target products and digital content to users. Recommender systems are particularly useful in environments that are characterized by numerous users who face a vast array of products to choose from. In such contexts, because of the large number of products, users are often uncertain about or unaware of products that might appeal to them, and there is considerable heterogeneity in user preferences for product attributes. Moreover, such environments are often constantly evolving, as new users and new items are added on a regular basis. Firms therefore use recommender systems to offer personalized suggestions to users.

Recommendation systems need to overcome various modeling and computational challenges to successfully predict preferences and recommend products. Such systems often operate on a sparse database in which each consumer rates only a few items and each product is rated or chosen by only a few customers. The paucity of data for most consumers implies that it is critical to borrow information from other consumers to predict the preferences of a given consumer (Ansari et al. 2000). The large number of products also poses a challenge in representing these items in terms of their underlying features. Such feature representations are often unavailable, or, at best, partially available, as considerable domain expertise is needed to manually supply detailed content descriptors for each product. Yet, a rich representation of products in terms of their attributes is crucial for properly modeling preference heterogeneity. Thus, many systems rely on some sort of automatic feature extraction based on textual data or user-generated content (UGC; Lops et al. 2011). Finally, recommender systems need to overcome various cold-start problems in dealing with new users or new items.

Apart from the above modeling challenges, typical recommendation contexts generate big-data problems stemming from data volume, variety, and veracity. Although personalization focuses on a given user or a given product, the large data volume that results from a massive user base and a vast product mix are critical

for recommendation success, as they facilitate the borrowing of information and enrich the representation of products. However, these also result in scalability challenges, especially when complex probabilistic representations are needed to fully capture the information content in the data. Moreover, automatic feature extraction based on online texts and product tags implies a curse of dimensionality that necessitates appropriate dimensionality reduction procedures. Thus, scalable methods that are capable of estimating probabilistic models containing many latent variables on large data sets of variegated forms are required.

In this paper, we propose a novel hybrid model-based recommendation framework that addresses these modeling and scalability challenges. Specifically, we construct a covariate-guided, heterogeneous supervised probabilistic topic model that synergistically uses product ratings, textual descriptions of products or user-generated product tags, and firm-provided product covariates to automatically infer the set of latent product features (topics) that not only summarizes the semantic content within the tags/words, but is also most predictive of user preferences. The firm-specified product attributes are used to guide the allocation of topics to products. The latent topics result in an automatic dimension reduction of the vast vocabulary underlying the textual descriptions, and the model infers heterogeneous user preferences over these latent topics. This yields a recommendation system that leverages preference heterogeneity over rich user-generated content representations in a seamless manner and is capable of handling various cold-start scenarios.

On the methodological front, our model extends the literature on supervised topic models (Blei and McAuliffe 2007) in several directions to accommodate the unique characteristics of the recommendation context. Recommendation data sets often have a complex dependency structure as a given user rates multiple products and each product is rated by multiple users. Thus, in our model, each product description (i.e., a document in the topic model) is associated with multiple product ratings given by different users. This is distinct from typical supervised topic models in which each document is rated by a single user—such models are more suitable for sentiment analysis of reviews, but are not rich enough to represent the preference heterogeneity that is crucial for successful recommendations. We therefore account for preference heterogeneity over the topics and explicitly take into account the nested structure of the data. We also use firm-specified product covariates to deviate from the restrictive exchangeability assumptions of supervised latent Dirichlet allocation (LDA) models. Furthermore, we develop a novel stochastic variational Bayesian (SVB) framework for the scalable estimation of our model.

We apply our model in the context of personalized movie recommendations and search. We show that our model generates much better predictions when compared with a benchmark model that uses only manually specified genre covariates. This illustrates the benefits that accrue from the rich feature representations derived from UGC and highlights that standard content descriptors such as the genre variables are not rich enough to flexibly capture the many reasons why certain movies appeal to particular users. We uncover a number of interesting insights about user preferences and about the semantic structure behind the movie tags. We then illustrate how the model is useful for the functioning of a recommender system. Specifically, we show how our model can generate product recommendations that are conditional on different information sets and how it can support a variety of personalized search tasks. These include generating a user-specific ranking of movies most similar to a given movie, or identifying and ranking movies relevant to a set of needs that a user specifies via keywords. Finally, because the set of movies a user rates may exhibit some degree of self-selection, we perform robustness checks using a Heckman (1979) selectivity correction (Ying et al. 2006), but find no significant differences in our results.

Our application can be considered a quintessential big-data example, as it simultaneously incorporates multiple facets of the Volume, Variety, Veracity, and Velocity framework that is used to characterize big-data situations (Sudhir 2016, Wedel and Kannan 2016). For instance, our application uses a large volume of ratings stemming from large sets of users and products. Also, the model uses a variety of data, including unstructured text and numbers, and summarizes the high-dimensional space of tags into a small set of latent topics. Moreover, our application showcases the challenges and opportunities of data veracity, in that data can be fused together from disparate sources, as the tags, ratings, and features can be gathered from different sets of users on various platforms. We show that our SVB algorithm, which leverages many novel computational features such as stochastic natural gradient descent and adaptive learning rates, yields estimation results in a fraction of the time needed for regular Markov chain Monte Carlo (MCMC) methods.

In summary, our research makes both methodological and managerial contributions. Methodologically, we develop a novel supervised topic model that incorporates a number of features that are relevant for recommendation and personalized search. In addition, we develop a new SVB framework that can be useful in a variety of big-data marketing scenarios. On the managerial front, our model can be used not only for generating insights about consumer preferences, but also for directly recommending products. As segmentation, targeting, and personalization are core marketing activities, our modeling and estimation approaches are immediately useful for marketers in a variety of product and service contexts.

The rest of this paper proceeds as follows. After a literature review in Section 2, we describe our data in Section 3. We develop our modeling framework in Section 4 and discuss scalable estimation in Section 5. Section 6 describes the results and managerial insights from our application. We then illustrate the use of our model for movie recommendation and personalized search in Section 7. We conclude by discussing the limitations of our model and by highlighting directions for further research.

2. Literature Review

Several research areas in marketing, statistics, and machine learning are relevant for our work on personalized recommendation systems in big-data settings. These include the literature on recommendation systems, the natural language processing (NLP) work on probabilistic topic models, and the ongoing research on scalable Bayesian inference in statistics and computer science. We succinctly review these areas below.

2.1. Recommendation Systems

A number of studies in marketing and computer science have developed memory-based or model-based methods for generating product recommendations. Prominent classes of recommendation algorithms include collaborative filtering, content filtering, and hybrid approaches that combine collaborative and content filtering.

Collaborative filtering systems (for a review, see Desrosiers and Karypis 2011) rely solely on user ratings or purchase data and do not utilize attribute information in making recommendations. User-based collaborative filtering recommends items to a user by leveraging the preferences of other users who are closest to the user. Similarly, item-based collaborative filtering identifies those products that are closest to a given product in terms of their appeal to customers and uses them for recommendations. More recent incarnations of collaborative filtering use matrix factorization (Koren and Bell 2011) of the user–item ratings matrix to uncover a limited number of latent factors that represent user preferences or unobservable product features. Despite their utility, collaborative filtering methods suffer from cold-start problems and cannot be used for new users or new items. Moreover, they do not provide any rationale for the recommendations they make.

Content filtering systems (for a review, see Lops et al. 2011), in contrast, use content information pertaining to an item to capture the drivers of preferences. Content is broadly defined and can take the shape of a set of product features that are either supplied or extracted from other data sources. Content-based systems can provide the underlying rationale for a recommendation and can, therefore, increase customer trust in the system. Content-based methods have additional advantages in that they can be used to predict preferences for new items based on their constituent features. However, manually coding a set of features that comprehensively describe an item can be difficult, especially when products are added on a continual basis or when dealing with a large number of products. Moreover, a complete description of a product requires many attributes, especially for experiential products such as movies. This can amplify the difficulty of data collection considerably, especially when domain experts are needed to specify the relevant attribute values.

Hybrid recommender systems integrate collaborative and content filtering to leverage the best features of both. Ansari et al. (2000) developed such a hybrid hierarchical Bayesian model to leverage user preference heterogeneity in making recommendations. Salakhutdinov and Mnih (2008) also discussed a related Bayesian probabilistic matrix factorization model. In such models, Bayesian shrinkage enables model-based collaborative filtering, whereas content is explicitly specified. A number of marketing scholars have made significant advances in studying hybrid recommender systems, including Ying et al. (2006), Bodapati (2008), Chung et al. (2009), and Chung and Rao (2012). In this paper, we continue in this tradition, but focus explicitly on leveraging automatic content representations obtained via probabilistic topic models to predict user preferences, and on the scalability challenges arising from big-data settings.

2.2. Natural Language Processing

The automatic content representation in our model relates to the NLP literature on probabilistic topic models for textual data (e.g., Blei et al. 2003, Blei and McAuliffe 2007, Tirunillai and Tellis 2014, Büschken and Allenby 2016). As outlined in the introduction, our work extends the supervised LDA model (Blei and McAuliffe 2007) in several directions to represent the unique requirements of recommendation systems. Whereas traditional supervised topic models are not suitable for personalized recommendations, our model uses a richer latent variable specification that allows for multiple ratings from different users for each document (movie) and accounts for user differences in their preference structure over the topics. Moreover, our model uses firm-specified product covariates to guide the allocation of topics to products, which is helpful for managing the cold-start problems and for improving recommendation performance.

The NLP literature has explored how user-generated tags can be used to infer feature representations. For instance, Michlmayr and Cayzer (2007) used tag co-occurrences to represent user preferences and employed simple string matching to establish a correspondence between preferences and product information. Firan et al. (2007) built tag-based user profiles for music recommendations. In their algorithm, individual liking/disliking is inferred from tag usage and frequencies of listened tracks. Szomszor et al. (2009) described a movie

recommendation system in which the similarity between the keywords of a movie and the previous tags a user had provided to other movies is used to make recommendations. As the authors acknowledged, such a system can be further improved by combining collaborative tagging with a more content-based strategy.

As such, de Gemmis et al. (2008) proposed a more sophisticated hybrid approach, in which user preference is learned from both product content and user-supplied tags, and the latter include not only the personal tags of the user, but also tags from other users on the same product—the so-called social tags (Nam and Kannan 2014). The pooling of tags across users is particularly important when the users generating the content have different levels of expertise in the product domain. We adopt the same social tagging strategy in setting up our movie recommendation model. Furthermore, topic models have been used previously in the context of product recommendations. Jin et al. (2005) used topics extracted from an unsupervised latent Dirichlet topic model to recommend products. In contrast, we use a more sophisticated supervised approach where the topics are informed simultaneously by the ratings, the user-generated tags, and other firm-supplied covariates.

2.3. Scalable Bayesian Inference

Finally, the statistical and machine learning literature on scalable Bayesian inference is relevant for the big-data setting of our application. Bayesian methods (Rossi et al. 1996) are particularly suited for recommendation problems, given the need to pool information across users in modeling heterogeneity and generating individual-level estimates of consumer preferences. MCMC methods are popular in summarizing the posterior distribution of latent variables and parameters, but can be slow in big-data situations because of the need for tens of thousands of iterations to achieve convergence. We therefore use variational Bayesian methods (Bishop 2006, Ormerod and Wand 2010, Dzyabura and Hauser 2011), which replace sampling with optimization, thus resulting in significant speed improvement. In particular, we leverage the state-of-the-art advances in stochastic variational methods (Hoffman et al. 2013, Tan 2015) to significantly enhance the speed and scalability of model inference for our movie recommender system.

Next, we describe the data context to facilitate an easier understanding of our model.

3. Data Description

We applied our model to MovieLens data (Harper and Konstan 2015) for movie recommendations. Our analysis uses the data set made available by MovieLens on August 6, 2015. The MovieLens system used a number of different mechanisms and interfaces over the span of the data to elicit ratings from users. For example, the movies that a user rated could be influenced by the mechanisms for searching, filtering, and ordering movies that were available at any particular point in time. Although users could rate unseen movies by relying on the linked Internet Movie Database (IMDb) descriptions of the movies, they could still decide not to rate a presented movie for a variety of reasons. Thus, the set of movies a user rated could reflect some degree of self-selection. Moreover, the exact consideration set of movies for a user is not observable, and so the data contain ratings for a nonrandom set of movies for each user.

The data set covers a time span from January 9, 1995, to August 6, 2015, and contains (1) movie ratings given by users on a 10-point scale ranging from 0 to 5 in 0.5-point increments, (2) textual tags applied to movies by the users, and (3) the title and genre information for each movie. The data set does not include any user demographics, and the movies are described by a set of 19 genres, where each movie can simultaneously belong to multiple genres. Not all users in the data set tagged every movie, so we aggregated all the tags applied to the same movie across users to construct a "bag-of-tags" description of the movie. Thus, in using the tags, we ignored the identities of the users who supplied the tags. It is important to note that using a bag-of-words description is not restrictive in our application, as there is no inherent sequential ordering to the tag applications, unlike when dealing with natural text (e.g., product reviews) where the semantic meaning of a given word depends critically on the sequence of words either preceding or succeeding it. To ensure an adequate number of tags for each movie, we focused on the set of 10,722 movies that received tags from at least four users and randomly selected 5,000 movies from this set for our analysis.

We used a number of preprocessing steps to clean the movie tags for statistical analysis. In particular, we converted the tags to lowercase to eliminate any redundancy that may arise from lowercase and uppercase versions of the same tag. We decided against tag stemming to facilitate easy understanding of the topics by readers and chose not to tokenize multiworded tags into space-separated words, because a tag as a whole is more meaningful than the individual words it comprises. To reduce vocabulary size, that is, the number of unique tags, to a manageable level, we also discarded all tags that were applied only once in the data. In addition, as our data contain well-formed tags and no free-flowing reviews or conversations, there was no need to remove stop words, as is typically done in textual preprocessing. These preprocessing steps resulted in a sample of 4,609 movies that were rated by 111,793 users. The total number of tag applications across all movies is 233,268, whereas the overall vocabulary size of 21,255 is much smaller, because the same tag can be applied to a given movie by multiple users. Compared with the 19 genres, this large

vocabulary has the potential to be a lot more expressive about the movie characteristics perceived by the users. The final data set contains 8,865,061 ratings across the users and movies.

Now we provide some summary statistics on the data. First, the proportions of the 19 movie genres in our sample are shown in Figure 1(a). We can see that drama, comedy, thriller, action, and romance are the top most represented genres, whereas film-noir is the least represented. Figure 1(b) shows a word cloud that reflects the most frequent tags applied to the movies. It is clear that many of these popular tags do not overlap with the 19 genres and include additional information about the theme, provenance, and cast of the movies, among other things. The diversity of the tags that is reflected in the word cloud highlights the importance of using social tagging to generate automated "attributes" beyond the traditional standard ones (i.e., genres) for describing and recommending products.

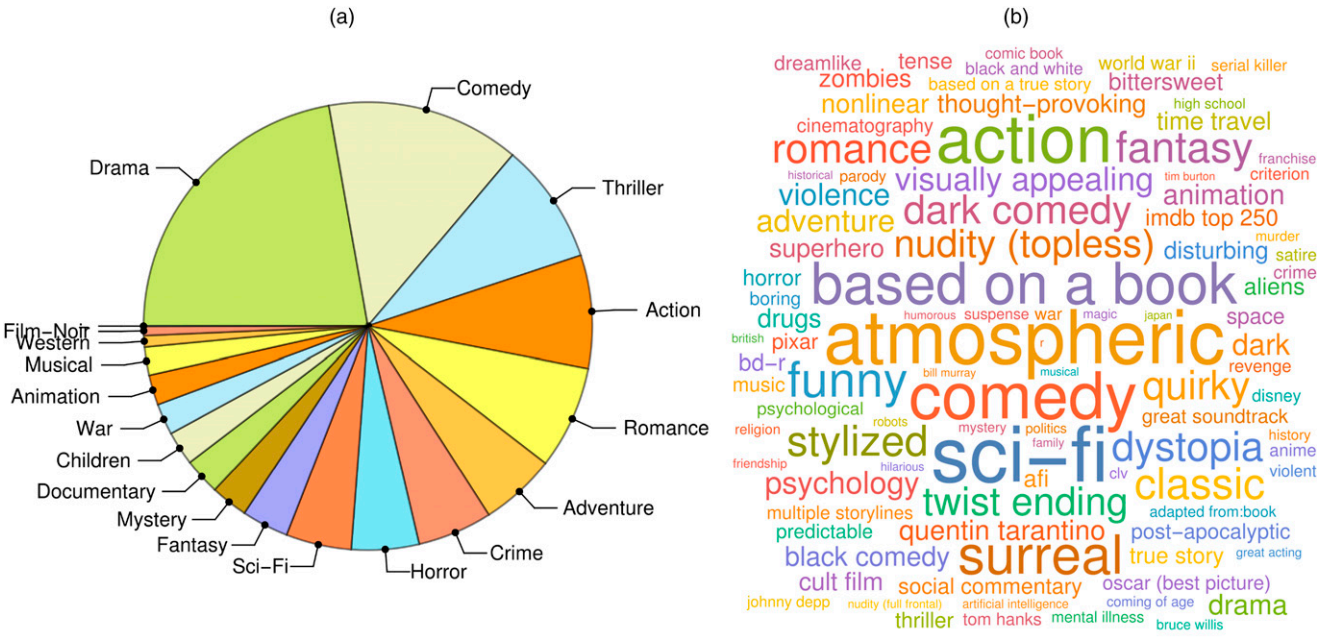
Figure 2 shows the histogram of the number of tags received by each movie. The median number of tags for a movie is 16, with a mean of 50 and a standard deviation of 114.9. It is interesting to note that the median number of tags is lower than 19, the number of prespecified genres within the data. The number of tags that are attached to a movie depends on the movie's popularity and on the time span for which it was part of the MovieLens database. Figure 3 shows word clouds of the tags for children's movies and romantic movies. It is clear from the figure that movies belonging to different genres have very different constellations of tags applied to them. For instance, children's movies show the frequent use of tags such as "animation," "funny," "Pixar," and "Disney." Moreover, it is heartening to note the lack of nudity in the set of children's movies.

The users in our data set differ significantly in the number and sets of movies they rated. The median number of movies a user rated is 43, and the mean is 79. As for the ratings, the mean across all observations is 3.57, with a standard deviation of 1.03, and the median rating is 4. We also computed the mean and standard deviation of the ratings received by each movie and, similarly, the mean and standard deviation of the ratings supplied by each user. These statistics indicates that there is considerable heterogeneity in the ratings at both the movie and user levels. This highlights the importance of accounting for individual differences in our modeling approach.

4. Model

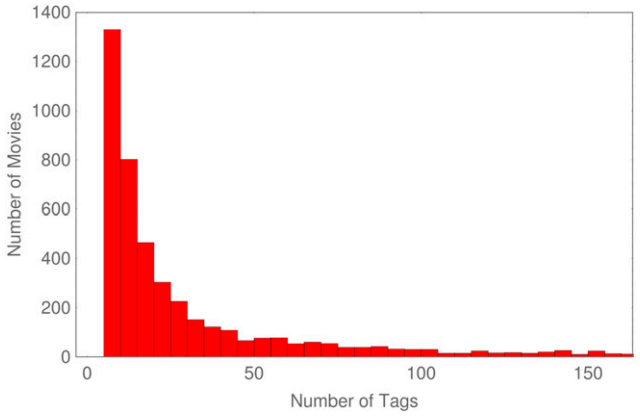
In this section, we develop a recommendation model that integrates multiple data modalities, including standard product covariates (e.g., movie genres), user-generated textual descriptions (e.g., tags), and user ratings. For ease of exposition, we take the set of products that a user has rated as given and discuss selectivity issues in the robustness checks. The overall model structure can be understood from the directed acyclic graph presented in Figure 4. We now describe the model in terms of the observed data, topic distributions, topic proportions, covariate guidance, tag applications, rating mechanism, and preference heterogeneity.

Figure 1. (Color online) Standard Movie Genres and User-Generated Tags



Proportion of Movie Genres | Word Cloud of Movie Tags

Figure 2. (Color online) Histogram of Number of Tags Received by Each Movie



4.1. Observed Data

We index the products (or documents/movies) by $d \in \{1, \ldots, D\}$ and the users by $i \in \{1, \ldots, I\}$, and represent the standard product attributes using a vector $\boldsymbol{x}_d$ that includes a constant term and binary indicators for the movie genres. We also assume that each product comes with a textual description that can take a variety of forms. For instance, this textual representation can be a bag-of-words agglomeration of user reviews for a product, or may come from web descriptions of the products. Alternatively, it can be composed from product tags. As depicted in Figures 1(b) and 3, in our application, this bag of words is a collection of user-generated movie tags.

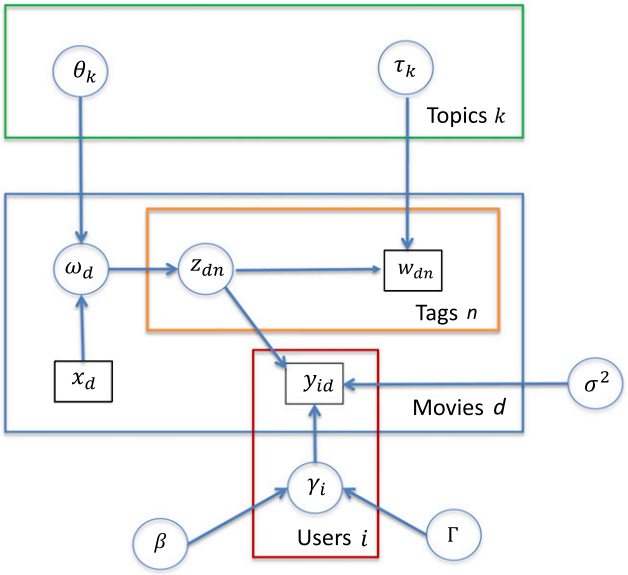
We represent the bag of tags for product d using a vector $\boldsymbol{w}_d = \{w_{dn}\}_{n=1}^{N_d}$, where w_{dn} is the nth token, or tag application, and N_d is the total number of tags applied to the product. Recall that a given product description can include multiple applications of the same tag from different users. The set of unique tags across all product descriptions is indexed by $v \in \{1, \ldots, V\}$, where V is the vocabulary size, or the total number of unique tags within the database. In addition to the product content, the database also contains product ratings y_{id} provided by the users as part of the normal operation of the recommender system. The matrix of product ratings, which represents the preferences of the users, is typically sparse, as only a fraction of products are rated by a given user, and a fraction of users provide ratings to a given product.

4.2. Topic Distributions

We assume that the tags can be summarized using a set of K "topics," where $K \ll V$. Such automatic summarization and dimensionality reduction via topics is critical for appropriately handling the large vocabulary size and the sparseness of the tag applications across products. A topic is a discrete probability distribution over the vocabulary. The kth topic is characterized by the probability vector $\boldsymbol{\tau}_k = \{\tau_{kv}\}_{v=1}^{V}$, where the element τ_{kv} indicates the probability with which the tag v occurs in that topic. The K topics differ in the probabilities τ_{kv} with which they generate a given tag v. In other words, a given tag has different probabilities of occurrence across the K topics and does not belong solely to a single topic. As these topic probability vectors lie in a $V-1$-dimensional simplex, we assume a symmetric Dirichlet prior, $\text{Dir}(\boldsymbol{\tau}_k|\eta)$, for the topic vector $\boldsymbol{\tau}_k$, where $\eta > 0$ is a scalar concentration parameter. The symmetric

Figure 3. (Color online) Word Clouds for (a) Children's and (b) Romantic Movies

Figure 4. (Color online) Directed Acyclic Graph for Main Model



Note. Latent variables are represented as circles and observed data as squares.

Dirichlet is a special case of Dirichlet distribution where all the K Dirichlet parameters are assumed to be equal to η. The marginal distribution for τ_{kv}, the vth element of $\boldsymbol{\tau}_k$, is a beta distribution with expectation $\mathbb{E}[\tau_{kv}] = 1/V$. Thus, assuming a symmetric Dirichlet prior is akin to using a discrete uniform distribution. This is appropriate for specifying the uncertainty over topic distributions as we do not possess prior knowledge about how different topics would favor one tag over another, and therefore our prior confirms with the principle of indifference (Keynes 1921).

4.3. Topic Proportions and Covariate Guidance

As each tag can come from any of the K topics with different probabilities, the bag of tags for a product (document) represents multiple topics. Thus, unlike a finite mixture specification in which each product description draws from a single topic, we assume that a product simultaneously belongs to all the topics with varying probabilities, thus yielding a mixed membership model (Erosheva et al. 2004). The probabilities with which the K topics are represented within the description for product d are given by the topic proportions vector, $\boldsymbol{\omega}_d = \{\omega_{dk}\}_{k=1}^K$. Figure 3 implies that the mix of tags applied to a movie depends on its genre. We therefore specify $\boldsymbol{\omega}_d$ to be a function of the product attributes $\boldsymbol{x}_d$. We use a Dirichlet regression to model this dependence, that is, $\boldsymbol{\omega}_d \sim \text{Dirichlet}(\exp(\boldsymbol{\Theta}\boldsymbol{x}_d))$, where $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_k\}_{k=1}^K$ is a matrix of regression coefficients for the K equations. The estimate of $\boldsymbol{\theta}_k$ represents how the product attributes impact the probability of a particular topic being present in the document. Therefore, the Dirichlet regression setup allows the standard product covariates $\boldsymbol{x}_d$ to guide the allocation of topics $\boldsymbol{\omega}_d$. Moreover, it enables borrowing of information within groups of movies having the same genres. Our use of such a conditionally exchangeable asymmetric Dirichlet distribution for the topic proportions is more general and flexible than typical LDA specifications that rely on a symmetric Dirichlet distribution for this task. Moreover, according to Wallach et al. (2009), such asymmetric priors result in more interpretable topics.

4.4. Tag Applications

We further associate a K-dimensional latent topic assignment vector $\boldsymbol{z}_{dn}$ with each tag n of the document d, such that the kth element of $\boldsymbol{z}_{dn}$ is a binary indicator that takes the value one with probability ω_{dk}. If the tag is assigned from topic k, then the actual tag is randomly drawn from the V-dimensional vocabulary with probability given by the topic distribution $\boldsymbol{\tau}_k$.

4.5. Rating Mechanism

We relate the ratings y_{id} to the tags using the empirical frequencies of different topics in the bag-of-words description of the product. Following Blei and McAuliffe (2007), we regress y_{id} on the unobserved average empirical frequencies $\bar{\boldsymbol{z}}_d$, where $\bar{\boldsymbol{z}}_d = (1/N_d)\sum_{n=1}^{N_d} \boldsymbol{z}_{dn}$, and allow each user to have her or his own regression coefficient $\boldsymbol{\gamma}_i$. Also, by regressing the ratings on the mean unobserved frequencies $\bar{\boldsymbol{z}}_d$, rather than on the topic proportions $\boldsymbol{\omega}_d$, we ensure that the ratings are determined by the topic frequencies that actually underlie the bag-of-tags description of the product. Such an approach is likely to yield topics that not only capture the semantic content of the tags, but are also most predictive of the user ratings and reflective of the standard product covariates. Using the topic proportions, instead, could result in specialization of topics, such that some of them only explain the ratings, whereas others exclusively summarize the tags. Therefore, the use of actual frequencies has the potential to improve the predictive ability of the model, which is central to the successful functioning of recommender systems.

4.6. Preference Heterogeneity

The user-specific coefficient $\boldsymbol{\gamma}_i$ reflects the extent to which the prevalence of different latent semantic dimensions (topics) within the textual description for a product matters in explaining the preference of user i. As the data contain multiple ratings from each user, we are able to properly account for sources of unobserved user heterogeneity, which is critical for capturing the diversity of user preferences on the latent topics. We model this heterogeneity via a normal distribution, $\boldsymbol{\gamma}_i \sim \mathcal{N}(\boldsymbol{\beta}, \boldsymbol{\Gamma})$, where $\boldsymbol{\beta}$ is the mean and $\boldsymbol{\Gamma}$ is the covariance of the population preference distribution.

The diagonal elements of $\mathbf{\Gamma}$ capture the variability in the preference parameters, and the off-diagonal elements reflect how the preferences for different latent topics covary across users. Our use of a population distribution allows us to leverage the Bayesian mechanism of borrowing information across users to explain the preference of a given user. The estimated coefficient for a user reflects a weighting between the individual's data and the population mean, such that the coefficients of those users with many ratings are mostly estimated from their data alone, whereas the coefficients for users with very few ratings are mostly shrunk toward the population mean. This is also beneficial in handling various cold-start scenarios involving new users.

4.7. Generative Process

We fuse together the tag applications, ratings, and product covariates to jointly uncover the latent topics that best predict user ratings and the preference parameters underlying these ratings. Given the above description, our model can be specified using the following generative process. Fixing the number of topics K, the Dirichlet parameter η, the Dirichlet regression parameters $\mathbf{\Theta}$, the population distribution parameters $\boldsymbol{\beta}$ and $\mathbf{\Gamma}$, and the regression error variance σ^2, our model generates product tags and their associated user ratings as follows:

1. Draw topic distribution for each topic, $\boldsymbol{\tau}_k \sim \text{Dir}(\eta)$.
2. Draw topic proportions for each product description, $\boldsymbol{\omega}_d \sim \text{Dir}(\exp(\mathbf{\Theta}\boldsymbol{x}_d))$.
3. For each tag application n belonging to product description d,
   (a) draw topic assignment $\boldsymbol{z}_{dn} \sim \text{Multinomial}(\boldsymbol{\omega}_d)$;
   (b) Draw tag $w_{dn} \sim \text{Multinomial}(\boldsymbol{\tau}_{z_{dn}})$.
4. For each user i who rated product d,
   (a) draw her or his preference parameters $\boldsymbol{\gamma}_i \sim \mathcal{N}(\boldsymbol{\beta}, \mathbf{\Gamma})$;
   (b) draw rating $y_{id} \sim \mathcal{N}(\bar{\boldsymbol{z}}_d'\boldsymbol{\gamma}_i, \sigma^2)$, where $\bar{\boldsymbol{z}}_d = (1/N_d)\sum_{n=1}^{N_d}\boldsymbol{z}_{dn}$.

Until now, we focused on specifying the model conditional on the set of products a user rated. However, if this set of movies is self-selected and nonrandom, it could be beneficial to explicitly accommodate the selection mechanism in the modeling framework. In our robustness checks, we model the "selection" stage and use the two-stage Heckman correction as in Ying et al. (2006). In particular, we assume that a user's decision to rate a product can be modeled via a heterogeneous binary probit regression as a function of the topic proportions of the product. We obtain these topic proportions from an unsupervised LDA model of the product tags, using the same number of topics as in the main model. The user-specific coefficients can then be used to compute the inverse Mills ratios of the probit (Greene 2003). In the second stage, we include the inverse Mills ratio for each rating observation in the main model as an additional covariate, using a straightforward modification of Step 4(b) above.

5. Posterior Inference via Stochastic Variational Bayes

We now focus on inference for our main model, as modifications to handle the selectivity correction via the addition of the inverse Mills ratio are straightforward. The full posterior distribution of our covariate-guided, heterogeneous supervised topic model can be written as

$$\begin{aligned} p(\boldsymbol{\omega}_{1:D}, \boldsymbol{z}_{1:D}, \boldsymbol{\tau}_{1:K}, \boldsymbol{\theta}_{1:K}, \boldsymbol{\gamma}_{1:I}, \boldsymbol{\beta}, \mathbf{\Gamma}, \sigma^2 \mid \boldsymbol{w}_{1:D}, \boldsymbol{y}_{1:I}, \boldsymbol{x}_{1:D}) \propto \\ p(\boldsymbol{\beta})p(\mathbf{\Gamma})p(\sigma^2)\prod_{k=1}^{K}\{p(\boldsymbol{\theta}_k)p(\boldsymbol{\tau}_k|\eta)\}\prod_{i=1}^{I}\{p(\boldsymbol{\gamma}_i|\boldsymbol{\beta}, \mathbf{\Gamma}) \\ \times \prod_{d\in\mathcal{D}_i} p\left(y_{id}|\boldsymbol{\gamma}_i, \boldsymbol{z}_d, \sigma^2\right)\} \times \prod_{d=1}^{D}\{p(\boldsymbol{\omega}_d|\boldsymbol{x}_d, \mathbf{\Theta}) \\ \times \prod_{n=1}^{N_d} p(w_{dn}|\boldsymbol{z}_{dn}, \boldsymbol{T})p(\boldsymbol{z}_{dn}|\boldsymbol{\omega}_d)\}, \end{aligned} \tag{1}$$

where $\boldsymbol{T} = \{\boldsymbol{\tau}_k\}_{k=1}^{K}$, and $\mathcal{D}_i$ denotes the set of movies rated by user i.

Because the normalizing constant cannot be computed in closed form, the posterior distribution is not available analytically, therefore necessitating approximate methods of inference. Marketers have traditionally used MCMC methods for summarizing the posterior distribution. MCMC methods involve iteratively sampling parameter values from the full conditional distributions, and inference is then made based on the sample of correlated draws. MCMC methods such as Gibbs sampling and the Metropolis–Hastings algorithm typically require tens of thousands of draws from the posterior. In big-data contexts such as movie recommendations that involve a large volume of data characterized by millions of ratings and high dimensionality resulting from massive numbers of user-supplied tags, MCMC methods are computationally intensive and take a long time to converge. We therefore use stochastic variational Bayesian approaches to approximate the posterior distribution. As this is the first application of SVB inference in the marketing literature, we briefly review these methods before deriving a specific instantiation for our model.

5.1. Variational Bayesian Inference

Suppose $p(\boldsymbol{\nu}|\boldsymbol{y})$ represents the posterior, $p(\boldsymbol{y}, \boldsymbol{\nu})$ denotes the joint distribution, and $p(\boldsymbol{y})$ is the normalizing constant for a generic Bayesian model. Variational Bayes (VB) methods approximate the intractable posterior $p(\boldsymbol{\nu}|\boldsymbol{y})$ with a simpler approximating distribution $q(\boldsymbol{\nu}|\boldsymbol{\lambda})$, called the variational distribution (Jordan et al. 1999, Bishop 2006, Ormerod and Wand 2010), that is indexed by a set of variational parameters $\boldsymbol{\lambda}$. In variational inference, we search over the space of variational distributions to find a member that is closest to the posterior distribution. The closeness between the approximating

distribution $q(\boldsymbol{\nu}|\boldsymbol{\lambda})$ and the posterior $p(\boldsymbol{\nu}|\boldsymbol{y})$ is measured by the Kullback and Leibler (1951) (KL) divergence,

$$\begin{aligned}\mathrm{KL}\left[q(\boldsymbol{\nu}|\boldsymbol{\lambda})\|p(\boldsymbol{\nu}|\boldsymbol{y})\right] &= \int q(\boldsymbol{\nu}|\boldsymbol{\lambda})\log\frac{q(\boldsymbol{\nu}|\boldsymbol{\lambda})}{p(\boldsymbol{\nu}|\boldsymbol{y})}d\boldsymbol{\nu}\\ &= \mathbb{E}_q\left[\log q(\boldsymbol{\nu}|\boldsymbol{\lambda})\right] - \mathbb{E}_q\left[\log p(\boldsymbol{y},\boldsymbol{\nu})\right]\\ &\quad + \log p(\boldsymbol{y}) \geq 0, \end{aligned} \tag{2}$$

where the equality holds if and only if $q(\boldsymbol{\nu}|\boldsymbol{\lambda}) = p(\boldsymbol{\nu}|\boldsymbol{y})$. As the last term $\log p(\boldsymbol{y})$ is a constant, minimization of the KL divergence with respect to $q(\boldsymbol{\nu}|\boldsymbol{\lambda})$ is equivalent to maximizing the evidence lower bound, or $\mathrm{ELBO}(\boldsymbol{\lambda}) = \mathbb{E}_q[\log p(\boldsymbol{y},\boldsymbol{\nu})] - \mathbb{E}_q[\log q(\boldsymbol{\nu}\,|\,\boldsymbol{\lambda})]$.

Because we do not know the posterior $p(\boldsymbol{\nu}\,|\,\boldsymbol{y})$ to begin with, we must place restrictions on the approximating variational distribution for the optimization to proceed. These restrictions structure the approximating distribution such that its functional form is either inferred automatically from the model structure or explicitly set by the analyst via the selection of a specific parametric family of distributions. The choice of the restrictions reflects the trade-off between the tractability and the richness of the approximation. In practice, mean-field restrictions are popular in handling conjugate models (Bishop 2006, Ormerod and Wand 2010), whereas fixed-form approximations are often applied to nonconjugate setups (Braun and McAuliffe 2010, Knowles and Minka 2011, Salimans and Knowles 2013, Wang and Blei 2013, Titsias and Lazaro-Gredilla 2014). In this paper, we develop a novel hybrid VB framework that combines structured mean-field and fixed-form approximations to estimate the movie recommendation model. As the speed of model output is crucial in the big-data recommendation context, we also use stochastic optimization with adaptive minibatch sizes and adaptive moment estimation to speed up the estimation, resulting in a novel stochastic VB algorithm. Our algorithm also includes various forms of parallelization that leverages the conditional independence structure of the model. In the following, we provide enough detail for the presentation of these methods to be self-sufficient and relegate additional technical aspects to the appendices.

5.2. Hybrid Variational Bayes

We use a structured mean-field approximation that mimics the dependency structure of the joint distribution and specify the variational distribution as follows:

$$\begin{aligned}&q(\boldsymbol{\omega}_{1:D}, \boldsymbol{z}_{1:D}, \boldsymbol{\tau}_{1:K}, \boldsymbol{\theta}_{1:K}, \boldsymbol{\gamma}_{1:I}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \sigma^2)\\ &\quad = q(\boldsymbol{\beta})\,q(\boldsymbol{\Gamma})\,q(\sigma^2)\prod_{d=1}^{D}\left\{q(\boldsymbol{\omega}_d)\prod_{n=1}^{N_d}q(\boldsymbol{z}_{dn})\right\}\\ &\qquad \times \prod_{k=1}^{K}\{q(\boldsymbol{\theta}_k)q(\boldsymbol{\tau}_k)\}\prod_{i=1}^{I}q(\boldsymbol{\gamma}_i). \end{aligned} \tag{3}$$

We assume a normal prior for the population mean $\boldsymbol{\beta}$, an inverse-Wishart prior for the population covariance $\boldsymbol{\Gamma}$, and an inverse-gamma prior for regression error variance σ^2. We also assume a normal prior $\mathcal{N}(\boldsymbol{\mu}_\theta, \boldsymbol{\Sigma}_\theta)$ for the Dirichlet regression parameters $\boldsymbol{\theta}_k$. All unknowns except for $\boldsymbol{\theta}_k$ imply a semiconjugate setup; thus, we can derive closed-form variational expressions for the conjugate components in Appendix A. For ease of exposition, Table 1 summarizes the prior and the variational distributions for all the model parameters.

To estimate the nonconjugate component $\boldsymbol{\theta}_k$, we specify a multivariate normal variational component (Titsias and Lazaro-Gredilla 2014) and develop a novel adaptive doubly stochastic variational Bayesian (ADSVB) method, detailed in Appendix B, to compute $q(\boldsymbol{\theta}_k) = \mathcal{N}(\boldsymbol{\mu}_{q(\theta_k)}, \boldsymbol{\Sigma}_{q(\theta_k)})$. Combining the updates for the mean-field components with the ADSVB scheme yields an iterative coordinate ascent algorithm that uses ADSVB approach to update $q(\boldsymbol{\theta}_k)$ in an inner loop within an outer loop that updates all other conjugate parameters listed in Table 1. Appendix A provides the updating details for this hybrid VB procedure. To further speed up inference for big-data settings, in the following discussion we develop a stochastic optimization version of our hybrid VB scheme.

5.3. Stochastic Optimization with Adaptive Minibatches

When fitting a complex model with many individual-level latent parameters to a big data set, the coordinate ascent procedure requires significant computation because of the need to iteratively update every latent variable, including those for every user, within each iteration. This creates a computational bottleneck, especially if the data contain a large number of users. Recent research has explored strategies to enhance speed via stochastic variational inference (Hoffman et al. 2013).

Recall that the goal of variational Bayesian estimation is to maximize the ELBO. In our main model, the ELBO has the following form:

$$\begin{aligned}&\mathrm{ELBO}\\ &= \sum_{i=1}^{I}\left(\mathbb{E}\left[\log p(\boldsymbol{\gamma}_i|\boldsymbol{\beta},\boldsymbol{\Gamma})\right] + \sum_{d\in\mathcal{D}_i}\mathbb{E}\left[\log p\left(y_{id}|\boldsymbol{z}_d,\boldsymbol{\gamma}_i,\sigma^2\right)\right]\right)\\ &\quad + \sum_{k=1}^{K}\left(\mathbb{E}\left[\log p(\boldsymbol{\tau}_k|\eta)\right] + \mathbb{E}\left[\log p(\boldsymbol{\theta}_k)\right]\right)\\ &\quad + \sum_{d=1}^{D}\left(\mathbb{E}\left[\log p(\boldsymbol{\omega}_d|\boldsymbol{x}_d,\boldsymbol{\Theta})\right] + \sum_{n=1}^{N_d}\mathbb{E}\left[\log p(\boldsymbol{z}_{dn}|\boldsymbol{\omega}_d)\right]\right.\\ &\quad \left. + \sum_{n=1}^{N_d}\mathbb{E}\left[\log p(w_{dn}|\boldsymbol{z}_{dn},\boldsymbol{T})\right]\right) + \mathbb{E}\left[\log p(\boldsymbol{\beta})\right]\\ &\quad + \mathbb{E}\left[\log p(\boldsymbol{\Gamma})\right] + \mathbb{E}\left[\log p(\sigma^2)\right] + H(q), \end{aligned} \tag{4}$$

Table 1. Model Parameters, Priors, and Variational Distributions

Model Parameter	Prior Distribution	Variational Distribution
Conjugate components		
$\boldsymbol{\omega}_d$	$\text{Dir}(\exp(\boldsymbol{\Theta}\boldsymbol{x}_d))$	$\text{Dir}(\boldsymbol{\zeta}_d)$
$\boldsymbol{z}_{dn}$	$\text{Multinomial}(\boldsymbol{\omega}_d)$	$\text{Multinomial}(\boldsymbol{\phi}_{dn})$
$\boldsymbol{\gamma}_i$	$\mathcal{N}(\boldsymbol{\beta}, \boldsymbol{\Lambda})$	$\mathcal{N}(\boldsymbol{\mu}_{q(\gamma_i)}, \boldsymbol{\Sigma}_{q(\gamma_i)})$
$\boldsymbol{\Lambda}$	$\text{IW}(\rho_\Lambda, \boldsymbol{R}_\Lambda)$	$\text{IW}(\rho_{q(\Lambda)}, \boldsymbol{R}_{q(\Lambda)})$
$\boldsymbol{\beta}$	$\mathcal{N}(\boldsymbol{\mu}_\beta, \boldsymbol{\Sigma}_\beta)$	$\mathcal{N}(\boldsymbol{\mu}_{q(\beta)}, \boldsymbol{\Sigma}_{q(\beta)})$
α	$\mathcal{N}(\mu_\alpha, \sigma_\alpha^2)$	$\mathcal{N}(\mu_{q(\alpha)}, \sigma_{q(\alpha)}^2)$
σ^2	$\text{IG}(a_{\sigma^2}, b_{\sigma^2})$	$\text{IG}(a_{q(\sigma^2)}, b_{q(\sigma^2)})$
Nonconjugate components		
$\boldsymbol{\theta}_k$	$\mathcal{N}(\boldsymbol{\mu}_\theta, \boldsymbol{\Sigma}_\theta)$	$\mathcal{N}(\boldsymbol{\mu}_{q(\theta_k)}, \boldsymbol{\Sigma}_{q(\theta_k)})$

where the expectation is taken with respect to the associated variational distributions, and the last term $H(q)$ is the entropy of all variational distributions in the model. Maximizing the ELBO requires computing the gradient of the objective to move the iteration in the direction of the steepest ascent; that is, in each VB iteration we have to calculate the variational parameters for $\boldsymbol{\gamma}_i$ for all users to obtain the gradient. When the data contain a large number of users, as is the norm in the recommendation context, this becomes time-consuming. Because the ELBO and its gradient both involve a sum of individual terms that are independent conditional on population parameters, the theory on stochastic optimization (Robbins and Monro 1951, Spall 2003) helps us accelerate this optimization by *randomly* sampling a subset of users (minibatch) in each iteration to calculate a *noisy* gradient to replace the exact gradient.[1] Such a stochastic gradient, albeit computationally inexpensive, is asymptotically unbiased and makes the objective function probabilistically converge to an optimum under proper regularity conditions.

Note that the updates for the user-specific variational parameters within the minibatch depend on the population-level variational parameters, which are invariant across individuals. The minibatch strategy discussed above can be further improved in speed and scalability by sampling smaller batches of users in the initial VB iterations and allowing the batch size to increase adaptively over the iterations till it includes all the users (the entire data set). Such speed gain stems from the fact that the population parameters in the early iterations are most likely far from their optimum; therefore, it is wasteful to use these very "wrong" parameters to update many other (individual-level) parameters. Instead, we just sample smaller batches with fewer individuals as the iteration starts. In this way we can quickly move the optimization toward the (noisy) right direction. The batch size is increased by sampling more individuals when the current batch no longer suffices to move the estimation toward an optimum, that is, when more precision and more information are required for further convergence. Eventually, the minibatch reaches the full size of the data, after which the iterations are continued till convergence.

In essence, the SVB method is an *adaptive* procedure that automatically determines the most appropriate batch size to use in a given iteration, resulting in a significant enhancement in the speed and scalability of the already fast variational Bayesian estimation that uses the full data set in every iteration. To implement such an adaptive strategy, it is important to have a rule regarding when and how to increase the batch size during the iterations. In current paper, we adopt the *ratio of path and progress* (*RPP*) criterion (Gaivoronski 1988, Tan 2015) and use it on the fly to determine whether to sample more individuals into the minibatch and by how many users. Appendix C provides more details regarding this adaptive strategy, which is a part of the SVB estimation procedure outlined in Appendix A.

In addition, we leverage the conditional independence structure of the model to parallelize our optimization algorithm. As the preference parameters of each sampled user within a minibatch are conditionally independent, given the population parameters, we parallelize the updates of these user-specific parameters within the minibatch. Second, we parallelize the updates of the variational parameters for the topic assignments, as these are conditionally independent across documents (movies), given the population and user level parameters. Last, we parallelize the updates of topic distributions for every topic, as the distributions are independent conditional on other model parameters.

Having described our SVB methods, we need to acknowledge certain caveats regarding their properties. As VB methods approximate the posterior, the quality of the resulting estimates, when compared with those from MCMC methods, depends on the richness of the variational distribution and how well it captures the dependency structure of the full posterior distribution. A limited set of results are available on the properties of variational inference in specific settings. You et al. (2014) and Luts and Ormerod (2014) found that mean-field VB estimates of the posterior mean for

a linear model are consistent. Hall et al. (2011) proved consistency and asymptotic normality for a Poisson mixed effects model, and Titterington and Wang (2006) showed that coordinate ascent variational inference results in a consistent estimator for the posterior mean in the context of a mixture of normals. However, these results are specific to particular models and approximations, so there is a need for greater theoretical analyses of both the convergence and the properties of estimates obtained via variational inference. It is also known that variational inference tends to underestimate posterior standard deviations, particularly when a fully independent mean-field variational specification is used. This could be a concern when theory testing is of prime importance. Some recent research focuses on the use of robustness methods (Giordano et al. 2017) and more expressive approximations such as structured mean-field methods and normalizing flows (Rezende and Mohamed 2016) to handle this aspect. Thus, we use a structured mean-field specification to better capture the dependency structure of the posterior. Prediction accuracy, however, is governed by the recovery of parameter means, and in contexts such as ours, in which prediction is paramount, variational inference offers a good alternative to MCMC procedures when the latter are computationally prohibitive.[2]

6. Results

6.1. Null Model and Holdout Data

We now compare our model to a null model that has a hierarchical Bayesian linear specification and uses the prespecified 19 movie genres as covariates. As each movie is represented in terms of multiple genres, the null model has an intercept and a set of 19 genre-specific coefficients for each user. These user-specific effects are assumed to come from a multivariate normal population distribution with a full covariance structure, yielding a sophisticated benchmark specification. A comparison of the results from the two models allows us to assess the predictive benefits that stem from the richer semantic representations made possible by the latent topics.

Furthermore, given the nonrandom nature of the set of movies that a user rated, we performed robustness checks for our proposed model using the previously described two-step Heckman selectivity approach. Because our data contain no prior information regarding a user's consideration set, we augmented the set of movies that the user rated with a random sample of 1,000 movies that the user did not rate, and estimated a hierarchical probit regression on the topic proportions obtained from a regular LDA. The selectivity correction did not significantly change our results, either qualitatively in terms of the inferred topics or quantitatively in terms of the model predictions.[3] We therefore concentrate on the results from our original specification without the selectivity correction, and refer to the results from the model with selectivity, briefly, when discussing the predictive power of different models.

We estimated both the proposed model and the genre-only null model using stochastic variational Bayesian methods. Details of the mean-field coordinate-ascent variational updates for the null are available upon request. With a convergence criterion of 10^{-6} on the ELBO, the variational Bayesian estimation on the null model finishes in 10 iterations with 585.9 seconds (0.16 hours), whereas the regular MCMC estimation on the same model with 5,000 runs takes 8.0 hours to complete.[4] Also, the mean parameter estimates obtained from SVB and MCMC estimation for the null are virtually indistinguishable. Given that SVB estimation significantly outperforms MCMC estimation in speed, when estimating the main model, which is far more complex than the null, we did not use MCMC estimation. We expect that MCMC methods will not be competitive in terms of the computational time as our main model contains a very large number of latent variables, including the multinomial topic indicators for each of the 233,268 tag applications and multivariate user-level coefficients for each of the 111,793 users. The use of data augmentation to sample these latent variables is likely to be time-consuming, especially given that MCMC estimation typically requires a much larger number of iterations to converge, in comparison with VB estimation.

To evaluate the predictive performance of the genre-only model and the main model, we split our data set into calibration and holdout samples. We estimated both models on the calibration data and made predictions on both data sets. To form the holdout data, we set aside eight movies per individual. This resulted in a total of 7,970,717 ratings in the calibration data and 894,344 ratings in the holdout data.

We estimated multiple versions of the proposed model that differ in the number of topics K. Based on model fit, predictive performance, and topic interpretability, as reported in Table 4, we settled on a model with 20 topics, and the results presented here are from this version. We estimated our model using both deterministic and stochastic VB methods. The deterministic VB method that uses the full data in every iteration takes 18,930 seconds (5.3 hours) and 275 iterations to converge, whereas the stochastic VB method with adaptive minibatch sizes takes only 5,464 seconds (1.5 hours) and 164 iterations to finish. Convergence is declared when the joint Euclidean norm on the population parameters changes by less than 10^{-4} between iterations. The substantial difference in computational times once again highlights the scalability benefits of stochastic variational inference in big-data settings. As the actual estimates do not vary between the VB and SVB estimations, we now report the results from the SVB approach. We begin with the qualitative insights and discuss predictive performance and model fit subsequently.

Table 2. Top Eight Tags Associated with Each Topic

Topic	Top eight words within each topic							
1	nudity	betamax	alternate reality	future	bd-r	soundtrack	drama	romance
2	predictable	silly	boring	adam sandler	acting	martial arts	bad plot	comic book
3	franchise	stupid	sequel	nudity	remake	bad acting	not funny	silly
4	visually appealing	johnny depp	musical	ghosts	tim burton	adapted:book	based on a book	gothic
5	less than 300 ratings	religion	lesbian	christianity	queer	jesus	gay	british
6	stylized	dreamlike	surreal	tim burton	weird	johnny depp	space	fantasy world
7	romance	high school	chick flick	musical	love	marvel	romantic comedy	teen movie
8	zombies	gore	horror	cult film	disturbing	dystopia	creepy	war
9	dialogue	revenge	based on a book	french	acting	crime	science fiction	action
10	sci-fi	advanture	surreal	time travel	nonlinear	mindfuck	visually appealing	complicated
11	history	politics	tense	crime	documentary	true story	biography	corruption
12	surreal	japan	anime	adventure	stylized	hayao miyazaki	hallucinatory	studio ghibli
13	adventure	dystopia	action	space	artificial intelligence	post-apocalyptic	aliens	sci-fi
14	comedy	funny	humorous	parody	hilarious	black comedy	satire	high school
15	surreal	dark comedy	mental illness	multiple story lines	social commentary	gay	r	quirky
16	violence	quentin tarantino	world war ii	revenge	martial arts	nonlinear	brad pitt	great acting
17	suspense	psychology	black and white	bd-r	psychological	noir thriller	dystopia	serial killer
18	pixar	fantasy	animation	disney	adventure	fairy tale	comic book	children
19	classic	drama	romance	true story	oscar best picture	historical	tom hanks	inspirational
20	imdb top 250	atmospheric	dark comedy	quirky	black comedy	thought provoking	violence	twist ending

6.2. Topical Insights

6.2.1. Topic Distributions. Our model yields topic distributions that are most predictive of the ratings. Table 2 shows the top eight tags within each of the 20 topics. These topics are arranged in ascending order of their population mean coefficients β. Thus, the most important topics are shown at the bottom of Table 2. It is interesting to see how certain tags naturally congregate to form meaningful topics. For example, Topic 18 is about American animation movies for children. These are usually fairy tales and adventure themed, and are often produced by Pixar and Disney. Topic 12 is about Japanese animation movies, many of which are made by Mr. Miyazaki of Studio Ghibli. His films are quite different from those "family fun" Disney or Pixar productions, and tend to have story lines and characters that are fantastic, surreal, and dream-like in nature (*Spirited Away*, *Howl's Moving Castle*). Topic 16 appears to be closely related to the director Quentin Tarantino, who uses violence and nonlinear plot lines, and has directed works related to World War II (*Inglourious Basterds*, starring Brad Pitt), martial arts (*Kill Bill*), and revenge (*Kill Bill*, *Pulp Fiction*). It is clear that the topics richly depict the semantic information pertaining to the theme, provenance, popularity, awards, and actors of a movie. Thus, they capture a much greater semantic terrain than what is possible using a set of genre dummies.

Table 3 reports the population mean and variance associated with each topic coefficient. The table shows that the movies associated with the last few topics have high ratings, on average. We also see that the heterogeneity associated with the user coefficients γ_i varies across the topics. The large magnitude of these cross-user variances indicates that users exhibit considerable heterogeneity in their sensitivities on different topics. These results exhibit face validity as the higher numbered topics contain characteristics that most users would consider to be desirable. For example, Topic 19 contains "oscar best pictures," "true story," and "inspirational," and Topic 20 includes "imdb top 250" and semantics related to dark humor, which often sets a high requirement for screenplay excellence. All of these qualities are considered desirable by most people, resulting in higher means and smaller standard deviations. In contrast, lower numbered topics contain more negative and polarizing semantics. For instance, although some may be fans of silly, sequel, alternative reality, or Adam Sandler movies from Topics 1, 2 and 3, many others might consider such movies as undesirable, as evidenced by the tags such as "predictable," "boring," "bad acting," or "not funny." Scanning the adjectives across the topics also suggests that characteristics such as "nudity" or "franchise" do not predict greatness in the eyes of users, whereas "dark humor" is generally appreciated. These traits could inform studios about evolving public tastes.

6.2.2. Top Movies Associated with a Topic. We can also identify the top movies associated with each topic using the topic proportions ω_d for movies. Specifically, for a given topic k, we can find movies with the largest proportions ω_{dk} on this topic. Because of the limit imposed by page width, Table 5 shows only the top three movies associated with each topic. It can be seen by juxtaposing Tables 2 and 5 that these identified top movies are consistent with the semantic content of the topic. For example, the top movies for Topic 16 feature violent, martial arts movies—two of them directed by Tarantino—that are highly consistent with the top tags for that topic in Table 2. Another example is from Topic 19, which includes Oscar-winning dramas based on inspirational historical backgrounds (*Forrest Gump*, *Million Dollar Baby*, and *Titanic*). Movie enthusiasts would note that, although the top movies associated with Topic 20 are critically acclaimed, they are of a different flavor than those from Topic 19. For example, *Black Swan*, *Pulp Fiction*, and *Dead Poets Society* owe their success to quirky scripts instead of historical grandeur or high production value.

6.2.3. Topic Proportions for a Given Movie. Focusing on a given movie and using its topic proportions ω_d, we can study how the bag-of-tags representation draws from the different topics. Figure 5 shows the topic proportions for *Forrest Gump*. We see that this movie draws heavily from Topic 19, followed by Topics 20 and 14. A look at the tags associated with these topics in Table 2 reveals their striking relevance in describing *Forrest Gump*, an Oscar-winning romantic and inspirational drama starring Tom Hanks, set in historical America from the 1950s to present day (Topic 19), and with the quirky protagonist Forrest, who engages in behaviors and dialogue that is funny in a satirical way (Topics 14 and 20).

This example shows that our model can (1) flexibly describe a movie using multiple topics and (2) provide a relative ranking of the topics related to the movie that yields interesting and informative insights. For instance,

Table 3. Population Coefficients and Variances Associated with Each Topic

Topic	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
β	1.43	1.96	2.25	2.47	2.52	2.53	2.88	3.05	3.11	3.37	3.48	3.57	3.67	3.79	3.91	4.10	4.18	4.26	4.41	4.44
Diag(Γ)	1.70	1.68	1.72	1.47	1.58	0.89	1.95	0.85	1.48	1.79	1.19	1.07	1.53	0.98	1.46	1.60	1.37	1.06	1.32	1.58

Table 4. Measures of Predictive Performance

	Genre only	M10	M20	M30	M40	M50	Heckman20
Calibration sample							
MAD	0.62	0.55	0.52	0.51	0.51	0.50	0.52
RMSE	0.79	0.72	0.69	0.68	0.67	0.67	0.70
Correlation	0.61	0.71	0.74	0.76	0.76	0.77	0.74
Predictive R^2	0.42	0.51	0.55	0.56	0.58	0.59	0.56
Holdout sample							
MAD	0.70	0.65	0.62	0.62	0.61	0.61	0.63
RMSE	0.89	0.84	0.82	0.81	0.81	0.81	0.82
Correlation	0.51	0.58	0.61	0.61	0.61	0.62	0.61
Predictive R^2	0.22	0.34	0.37	0.37	0.38	0.38	0.36
Estimation time (hours)	0.16	0.99	1.52	2.89	4.16	5.83	1.71

although *Forrest Gump* has some comical characteristics found in Topics 14 and 20, most users would appreciate it for its inspirational features, described by Topic 19. Just because a movie is most related to a topic does not mean that it is married entirely to that topic. This flexible feature of our recommender system stems from the underlying mixed-membership model that governs the recommendations.

We see from the above that our model is capable of generating deep qualitative insights about the underlying drivers for movie preference. Such information is highly valuable in a recommender system, as it can be used to explain why a particular product is being recommended, something that is crucial for engendering trust.

6.3. Predictive Performance

We now discuss how well our model performs in predicting preferences. Table 4 presents the predictive performance measures for the genre-only null model and for the different versions of the proposed model, on both the calibration and the holdout data sets.

We report the mean absolute deviation (MAD), the root mean squared error (RMSE), and the correlation between the actual and the predicted ratings. In addition, the table also reports the predictive R^2 for these models. The column "Genre only" refers to the null model. The remaining columns refer to the variants of the proposed model. These versions differ in the number of topics, for example, "M10" indicates a 10-topic main model. From the comparison we can see that all the different versions of the covariate-guided, heterogeneous supervised topic model significantly outperform the genre-only model, even though the null is fairly sophisticated in its treatment of user heterogeneity. It is interesting to note that even the main model with just 10 topics does better than the null model with 19 genres. We also note that, in our case, adding more topics does not improve predictions much, but increases the computational time significantly. Of all the variants, the 20-topic version offers the best trade-off among model complexity, predictive performance, and topic interpretability. Therefore, we only report results from the 20-topic main model. In addition, the last column presents the statistics for our model with the selectivity correction. It is clear that accounting for selectivity using the available data does not improve predictions.

Table 6 compares the predicted ratings with the actual ratings to gauge the quality of the product recommendations. Users of the recommender system are interested in identifying good movies that conform with their taste; thus, suggestions from our model should correspond well with the good movies (i.e., those with high ratings) in the database. To assess this aspect, we divided the observations within our calibration and holdout samples into three groups (low, medium, and high) based on the one-third and two-third percentiles of the predicted ratings. We then computed the proportions of observations within each of these three groups that have low (actual ratings from 0 to 2), medium (actual ratings from 2.5 to 3.5), and high (actual rating from 4 to 5) ratings. For example, for the M20 model, the entry 0.929 of the high–high cell for the calibration sample indicates that 92.9% of the observations that the model predicts to have high ratings indeed have true ratings between 4 and 5. It is clear from the table that our proposed model predicts much better than the genre-only null model in each of the three groups, on both the calibration and holdout samples. We also see that the holdout predictions

Figure 5. (Color online) Topic Proportions for *Forrest Gump*

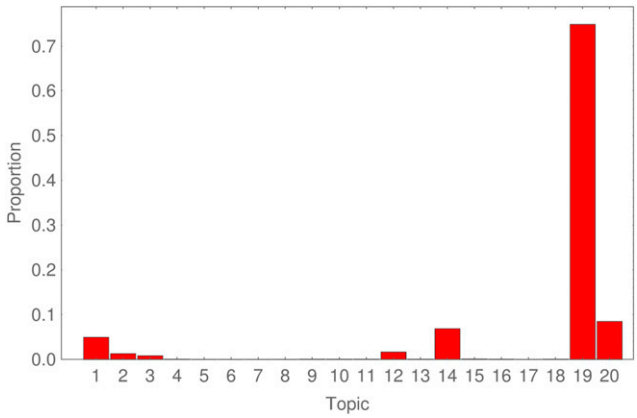


Table 5. Top Three Movies Associated with Each Topic

Topic	Movies		
1	*The Dreamer* (2003)	*Paris, je t'aime* (2006)	*Last Tango in Paris* (1972)
2	*Dinner for Schmucks* (2010)	*You Don't Mess with the Zohan* (2008)	*Due Date* (2010)
3	*American Pie 2* (2001)	*American Pie* (1999)	*The Waterboy* (1998)
4	*Alice in Wonderland* (2010)	*Dark Shadows* (2012)	*Mamma Mia!* (2008)
5	*The Passion of the Christ* (2004)	*Shelter* (2007)	*C.R.A.Z.Y.* (2005)
6	*The Holy Mountain* (1973)	*The Meaning of Life* (1983)	*The Hudsucker Proxy* (1994)
7	*Heathers* (1989)	*Bring It On* (2000)	*Romy and Michele's High School Reunion* (1997)
8	*Day of the Dead* (1985)	*Hostel* (2005)	*Martyrs* (2008)
9	*Robin Hood: Men in Tights* (1993)	*Christmas Vacation* (1989)	*Sapceballs* (1987)
10	*Primer* (2004)	*Mr. Nobody* (2009)	*Timecrimes* (2007)
11	*Frost/Nixon* (2008)	*The King's Speech* (2010)	*Moneyball* (2011)
12	*Spirited Away* (2001)	*My Neighbor Totoro* (1988)	*Howl's Moving Castle* (2004)
13	*Edge of Tomorrow* (2014)	*Mad Max: Fury Road* (2015)	*Star Wars: Revenge of the Sith* (2005)
14	*Superbad* (2007)	*Anchorman: The Legend of Ron Burgundy* (2004)	*Old School* (2003)
15	*Magnolia* (1999)	*Clerk* (1994)	*Shutter Island* (2010)
16	*Inglourious Basterds* (2009)	*Yip Man* (2008)	*Kill Bill: Volume 1* (2003)
17	*Seven* (1995)	*Rear Window* (1954)	*No Country for Old Men* (2007)
18	*Toy Story 3* (2010)	*Up* (2009)	*Beauty and the Beast* (1991)
19	*Forrest Gump* (1994)	*Million Dollar Baby* (2000)	*Titanic* (1997)
20	*Pulp Fiction* (1994)	*Black Swan* (2010)	*Dead Poets Society* (1989)

are slightly worse than those in the calibration sample, as expected.

7. Movie Recommendations and Personalized Search

We now focus on how to use our model to make personalized recommendations for specific users, conditional on different information sets. We also show how our model can be used to support personalized search based on user queries involving movies and keywords.

7.1. Cold-Start Recommendations

As new movies or users are added on a continual basis to the recommender system, it is important to show how our model can generate recommendations in different cold-start scenarios. For example, when a new user $\tilde{i}$ appears, without any prior information about her or him, we can use the variational estimates of the population distribution to predict the user's rating on any given movie in the data set with $y_{\tilde{i}d} = \bar{\boldsymbol{\phi}}_d' \boldsymbol{\mu}_{q(\beta)}$, where $\bar{\boldsymbol{\phi}}_d = (1/N_d)\sum_{n=1}^{N_d} \boldsymbol{\phi}_{dn}$ and $\boldsymbol{\mu}_{q(\beta)}$ represents population mean preference, and we recommend the highest rated movies accordingly. Similarly, when a new movie $\tilde{d}$ comes out on the market, it takes some time for user-generated content to become available. However, genre information $\boldsymbol{x}_{\tilde{d}}$ is readily available and can be used to compute the expected topic proportions with $\bar{\boldsymbol{\omega}}_{\tilde{d}} \propto \exp(\boldsymbol{M}_\theta \boldsymbol{x}_{\tilde{d}})$, where $\boldsymbol{M}_\theta = \{\boldsymbol{\mu}_{q(\theta_k)}\}_{k=1}^{K}$. We can then predict user i's rating for this new movie with $y_{i\tilde{d}} = \bar{\boldsymbol{\omega}}_{\tilde{d}}' \boldsymbol{\mu}_{q(\gamma_i)}$. Finally, to predict a new user $\tilde{i}$'s rating on a new movie $\tilde{d}$ (i.e., the case with least information), we can simply use the variational estimates of population preference with firm-provided product attributes to produce the prediction, $y_{i\tilde{d}} = \bar{\boldsymbol{\omega}}_{\tilde{d}}' \boldsymbol{\mu}_{q(\beta)}$.

7.2. Recommendations Conditional on Past Ratings

Now we discuss recommendations for users or items that are already part of the database. The simplest form of conditional recommendations are those based solely on the past ratings of a user and are made using movies that are part of the estimation data set but have not yet been rated by the user. The expected rating y_{id}^* for user i and movie d can be computed using the variational distributions for $\boldsymbol{\gamma}_i$ and $\boldsymbol{z}_d$, that is, $y_{id}^* = \bar{\boldsymbol{\phi}}_d' \boldsymbol{\mu}_{q(\gamma_i)}$. The expected rating can be computed for all the movies that the user has not rated, and the top-most movies, in terms of expected ratings, can then be recommended to the user.

Table 7 shows the 10 most preferred movies, identified using the above procedure, for three randomly chosen users in our data. The numerical entries under each movie indicate the probability with which the movie occupies that particular rank, given the estimated uncertainty on individual preferences.[5] As we can see, the recommendation set differs considerably across the three

Table 6. Predicted vs. Actual Ratings

	Actual rating					
	Genre-only model			M20 model		
Predicted rating	Low	Medium	High	Low	Medium	High
Calibration sample						
Low	0.44	0.37	0.19	0.57	0.36	0.07
Medium	0.13	0.37	0.50	0.09	0.40	0.51
High	0.08	0.14	0.78	0.01	0.06	0.93
Holdout sample						
Low	0.31	0.38	0.31	0.41	0.41	0.18
Medium	0.11	0.32	0.57	0.09	0.34	0.57
High	0.11	0.16	0.73	0.02	0.09	0.89

Table 7. Top Recommended Movies with Rank Probabilities

	USER 1		USER 2		USER 3	
Rank	Movie Title	Probability	Movie Title	Probability	Movie Title	Probability
1	*Transformers: Revenge of the Fallen* (2009)	0.46	*Pulp Fiction* (1994)	0.70	*Casablanca* (1942)	0.87
2	*Armageddon* (1998)	0.23	*Grand Budapest Hotel, The* (2014)	0.41	*Sunset Blvd.* (1950)	0.53
3	*Pearl Harbor* (2001)	0.35	*Inglourious Basterds* (2009)	0.31	*Citizen Kane* (1941)	0.39
4	*Transformers: Dark of the Moon* (2011)	0.18	*Django Unchained* (2012)	0.29	*Rear Window* (1954)	0.39
5	*Star Wars: The Phantom Menace* (1999)	0.23	*City of God* (2002)	0.22	*Third Man, The* (1949)	0.34
6	*Con Air* (1997)	0.31	*American Beauty* (1999)	0.27	*Lawrence of Arabia* (1962)	0.22
7	*Star Wars: Attack of the Clones* (2002)	0.20	*Lock, Stock & Two Smoking Barrels* (1998)	0.26	*City Lights* (1931)	0.34
8	*Transformers* (2007)	0.20	*Kill Bill 1* (2003)	0.22	*Maltese Falcon, The* (1941)	0.29
9	*G.I. Joe: The Rise of Cobra* (2009)	0.24	*Snatch* (2000)	0.38	*12 Angry Men* (1957)	0.21
10	*Terminator Salvation* (2009)	0.19	*American History X* (1998)	0.36	*To Kill a Mockingbird* (1962)	0.45

users, reflecting their different tastes. In particular, User 1 prefers visually stunning, large-production action films. User 2, instead of liking large Hollywood productions, prefers movies with quirky story lines and leading characters with unusual logic, and appears to be a big fan of director Quentin Tarantino's films. User 3 clearly prefers classics. These results highlight the importance of capturing preference heterogeneity to support personalized recommendations. In addition, the rank probabilities also vary across the users. Such variation comes from the fact that the three users provided different numbers of ratings (11, 33, and 76 ratings, respectively) in our calibration data set.

Remember that our model accounts for the topic mix and textual description of every film. Therefore, it can also support personalized search and can generate personalized rankings of movies, conditional on user queries that involve movie titles or movie keywords. Such personalized search and ranking based on additional user input are useful in that users may express different interests at different times, even though their latent movie preferences may remain largely unchanged. Thus, the search or browsing context can be leveraged to improve recommendation quality. We now show how the tags can be used to identify movies that are similar to a queried movie.

7.3. Movies Similar to a Given Movie

Item-based collaborative filtering algorithms identify the products that are closest to a given product in their appeal to customers. This is usually done using solely the ratings matrix, and, therefore, this approach suffers from the inability to provide an explanation regarding why a particular product is being recommended.

In contrast, our model can be leveraged to compute meaningful distances between movies based on their topic proportion vectors. Although many different distance metrics can be used for this task, here we use the Hellinger distance (Nikulin 2001) to compute the similarity between two movies, d and d', based on their topic proportions, $\boldsymbol{\omega}_d$ and $\boldsymbol{\omega}_{d'}$, respectively. The Hellinger distance satisfies the triangle inequality and is defined as

$$H(d,d') = \sqrt{\frac{1}{2}\sum_{k=1}^{K}\left(\sqrt{\omega_{dk}} - \sqrt{\omega_{d'k}}\right)^2}. \tag{5}$$

The use of topic proportions means that both ratings and textual content are utilized in computing closeness between movies, as the topic proportion is inferred by taking into account the tags, the genre covariates, and the ratings. This is therefore different from relying solely on either the movie ratings or the content of movies to compute similarity.

Table 8. Five Movies Most Similar to Given Movie

	Movies		
	Pulp Fiction (1994)	*The Dark Knight* (2008)	*The Lord of the Rings* (2003)
Similar movies	*Inglourious Basterds* (2009)	*Léon: The Professional* (1994)	*The Lord of the Rings* (2001)
	Kill Bill: Volume 1 (2003)	*The Dark Night Rises* (2012)	*The Hobbit* (2013)
	Reservoir Dogs (1992)	*The Prestige* (2006)	*WALL·E* (2008)
	Kill Bill: Volume 2 (2004)	*Lucky Number Slevin* (2006)	*Star Wars* (1980)
	Django Unchained (2012)	*The Game* (1997)	*Indiana Jones* (2008)

Table 9. Top Recommended Movies with Rank Probabilities Based on Search of Movie

	USER 1		USER 2		USER 3	
Rank	Movie Title	Probability	Movie Title	Probability	Movie Title	Probability
1	*From Dusk Till Dawn* (1996)	0.50	*Inglourious Basterds* (2009)	0.68	*Once Upon a Time in the West* (1968)	0.89
2	*Grand Budapest Hotel, The* (2014)	0.31	*Django Unchained* (2012)	0.36	*Grand Budapest Hotel, The* (2014)	0.37
3	*Reservoir Dogs* (1992)	0.29	*Kill Bill 1* (2003)	0.41	*Inglourious Basterds* (2009)	0.32
4	*Kill Bill 1* (2003)	0.24	*Grand Budapest Hotel, The* (2014)	0.31	*Reservoir Dogs* (1992)	0.27
5	*Django Unchained* (2012)	0.29	*Kill Bill 2* (2004)	0.28	*Kill Bill 1* (2003)	0.36
6	*Kill Bill 2* (2004)	0.34	*Reservoir Dogs* (1992)	0.39	*From Dusk Till Dawn* (1996)	0.38
7	*Sin City* (2005)	0.32	*Sin City* (2005)	0.44	*Django Unchained* (2012)	0.75
8	*Inglourious Basterds* (2009)	0.48	*Once Upon a Time in the West* (1968)	0.46	*Kill Bill 2* (2004)	0.72
9	*Once Upon a Time in the West* (1968)	0.58	*Drive* (2011)	0.64	*Drive* (2011)	0.79
10	*Drive* (2011)	0.46	*From Dusk Till Dawn* (1996)	0.52	*Sin City* (2005)	0.73

Table 8 illustrates three examples of five movies that are most similar to a given movie, based on the Hellinger distance. It is interesting to see that the movies deemed similar to *Pulp Fiction* mostly feature the directorial talent of Quentin Tarantino. The movies most similar to *The Dark Knight* (2008), include, in addition to its sequel in 2012, thrillers with complicated plots and elements of suspense. Finally, good (nonpersonalized) recommendations for those looking for something similar to *The Lord of The Rings* (2003), appear to be science-fiction and fantasy films that have large production values. It is apparent that the above set of results exhibit high face validity.

The ability to identify nearest neighbors in content space while simultaneously accounting for user preferences is highly beneficial in the day-to-day operation of a recommender system, as it enables the capability of suggesting additional movies that are similar to a movie the user queries about. Below we apply the movie similarity measures to implement personalized search based on user queries involving movie title or movie keywords.

7.4. Movie-Based Personalized Search

When a user actively looks for movies that are similar to a given movie, either by typing in the name of the movie or by browsing the description of the movie, we can leverage this extra information to obtain *personalized* rankings of movies that are most similar to the searched movie. For instance, with a search of the movie *Pulp Fiction*, we can use the Hellinger distance to identify the most similar movies, that is, the 10 movies shown in Table 9. These items constitute the recommendation set of relevant movies. We can then compute the predicted ratings, $y^*_{id} = \bar{\boldsymbol{\phi}}'_d \boldsymbol{\mu}_{q(\gamma_i)}$, for each user on the 10 movies. As users differ in their preference parameter $\boldsymbol{\mu}_{q(\gamma_i)}$, the ranking of the 10 movies can be personalized.

The personalized ranking and top recommended movies of our three users, conditional on their search for *Pulp Fiction* and on their preference parameters, are shown in Table 9. We can see that out of the 10 relevant movies, *From Dusk Till Dawn*, a vampire-killing film, perhaps offers the most amount of nonstop action to fit the preferences of User 1. The top recommendation for User 2 is *Inglourious Basterds* (2009), a Tarantino film. User 3's top recommendation is *Once Upon a Time in the West* (1968), the oldest and most classic film within the set. Although the movie name is a special type of search keyword, we now show how other keywords from the movie tags can be used to support personalized search.

7.5. Keyword-Based Personalized Search

When a user searches movies using a list of keywords from the movie tags, the keywords can be grouped to form a new "document" that describes a "movie" the user is interested in. This new document can then be used to make recommendations. Specifically, for every searched keyword, we can take the K-element probability vector $\boldsymbol{\tau}_v$, which gives the probability of the keyword v in each of the topics, and aggregate these vectors across all the keywords in the query and normalize them to get the topic proportions for the new "document." The Hellinger distance can then be used to determine the set of most similar movies, from which we apply the preference parameter to calculate a personalized ranking for the user.

For instance, if a user searches the two keywords "heartwarming" and "inspirational," the model can generate a set of 10 movies that are closest to the new document consisting of the two tags. Table 10 shows the relevant set of "heartwarming, inspirational" films and their personalized ranking for our three users. The ranking differs across the users because it is driven by the variation in their preferences. For instance, *Braveheart* (1995), a large-production action movie, is the top most recommendation for User 1, who is the "action enthusiast." User 2's top recommendation is *Into the*

Table 10. Top Recommended Movies with Rank Probabilities Based on Search of Keywords

	USER 1		USER 2		USER 3	
Rank	Movie Title	Probability	Movie Title	Probability	Movie Title	Probability
1	*Braveheart* (1995)	0.57	*Into the Wild* (2007)	0.62	*Ben-Hur* (1959)	0.83
2	*Forrest Gump* (1994)	0.52	*Saving Private Ryan* (1998)	0.52	*Fiddler on the Roof* (1971)	0.63
3	*Saving Private Ryan* (1998)	0.58	*Forrest Gump* (1994)	0.43	*Saving Private Ryan* (1998)	0.57
4	*Dances with Wolves* (1990)	0.45	*Braveheart* (1995)	0.26	*Braveheart* (1995)	0.46
5	*Secondhand Lions* (2003)	0.24	*Cider House Rules, The* (1999)	0.27	*Dances with Wolves* (1990)	0.66
6	*Amistad* (1997)	0.34	*Amistad* (1997)	0.46	*Forrest Gump* (1994)	0.49
7	*Cider House Rules, The* (1999)	0.51	*Dances with Wolves* (1990)	0.84	*Cider House Rules, The* (1999)	0.52
8	*Into the Wild* (2007)	0.66	*Ben-Hur* (1959)	0.82	*Into the Wild* (2007)	0.72
9	*Fiddler on the Roof* (1971)	0.58	*Secondhand Lions* (2003)	0.64	*Amistad* (1997)	0.70
10	*Ben-Hur* (1959)	0.55	*Fiddler on the Roof* (1971)	0.70	*Secondhand Lions* (2003)	0.68

Wild (2007), which features a young college grad who shuns the material world, cuts off communication with his family, lives off the land in Alaska, and eventually dies from food poisoning. Consistent with User 2's preference, this is a small production film with unusual protagonist actions and story line. *Ben-Hur* (1959) is the top recommendation for User 3. This movie is an epic historical drama, which could please our "classics enthusiast."

In summary, the results show our model cannot only predict the ratings well, but can also yield a number of qualitative insights regarding the structure of user preferences and the high-dimensional semantic space that characterizes user perceptions of movies. Moreover, our model can be useful in supporting personalized movie recommendations based on different information sets underlying various search scenarios and recommendation contexts.

8. Conclusion

In this research, we contribute to the literature on recommendation systems by developing a novel covariate-guided, heterogeneous supervised topic model that leverages numerical ratings, texts, and standard product attributes to make recommendations based on latent topics. The topics are inferred from both the tag vocabulary and the user ratings, thereby enhancing the predictive ability of the model, and our use of crowd-sourced tags alleviates the often onerous need for firm-provided product attributes. We developed a new stochastic variational Bayesian approach for scalable estimation and used it to estimate our model on a large data set of movie ratings and semantic tags. We show that our recommendation model generates much better predictions than the benchmark model and yields a number of interesting insights regarding user preferences, something that is not possible with the benchmark model. Our SVB methodology ensures a fast estimation, thereby making our approach useful in actual recommendation contexts.

We show how our modeling framework can produce targeted recommendations for users and support personalized search based on movie similarity or relevance to a set of specific keywords. This is possible as the topic proportions can be used to measure the perceptual distance between movies within the high-dimensional semantic space of user-generated tags. When combined with user-level preference parameters, the perceptual distance can yield a unique ordering of the relevant movies for each user, thus resulting in personalized search rankings.

On the methodology front, we developed a hybrid stochastic variational Bayesian framework for scalable inference in models that involve both conjugate and nonconjugate components. Although we showcased our estimation framework in the context of product recommendations, it can be used in a variety of big-data settings. The rapid growth in data volume, data variety, and crowdsourcing has opened new opportunities for deeper learning about customer preferences and for using predictions from such data for marketing actions. Novel and complex marketing models such as ours that include many latent variables are needed to fully capitalize on the information contained in these information-rich big-data scenarios. Scalable inference therefore becomes one of the major challenges in big-data and big-learning environments, and traditional approaches in Bayesian inference based on regular MCMC methods do not scale well for practical big-data applications. In contrast, by building on state-of-the-art advances in variational Bayesian inference, we developed a stochastic VB algorithm that offers a versatile solution to the scalability and computational challenges in estimating complex marketing models.

Although we focused on stochastic VB methods in the current paper, other approaches such as stochastic gradient-based MCMC and the divide-and-conquer approaches to MCMC are being actively investigated to tackle scalability issues. We leave a thorough comparison of these emerging methodologies for future research. Finally, although we explored the movie recommendation context, our model can also be applied to other situations where text data prevail. These include

matching markets such as the job hunting market, where candidates can be matched with firms based on the text within their resumes and firm preferences. We look forward to exploring these and other applications with our framework.

Acknowledgments

The authors are grateful to the MovieLens team at the University of Minnesota, for providing the invaluable data to our inquiries. The authors also thank the editor, the associate editor, the two anonymous reviewers, as well as the participants of seminars and conferences in which this work has been presented, for their constructive comments.

Appendix A. Stochastic Variational Bayesian Updates

We estimate the proposed model via SVB with adaptive minibatch sizes. We include the inverse Mills ratios χ_{di} in the rating equation to address selectivity issue such that $y_{id} \sim \mathcal{N}(\bar{\boldsymbol{z}}_d'\boldsymbol{\gamma}_i + \alpha\chi_{id}, \sigma^2)$. In each iteration t, we sample a subset of users $\mathcal{S}^{(t)}$ using the RPP strategy discussed in Appendix C. Let $\mathcal{D}_s = \cup_{i\in\mathcal{S}^{(t)}}\mathcal{D}_i$ denote the documents rated by the sampled individuals in the current iteration. We first update the variational parameters for $d \in \mathcal{D}_s$. Specifically, we calculate the variational parameters for topic proportions ω_{dk}, for all $d \in \mathcal{D}_s$ and $k = 1\ldots K$, by

$$\zeta_{dk} = \exp\{\boldsymbol{x}_d'\boldsymbol{\mu}_{q(\theta_k)} + \frac{1}{2}\boldsymbol{x}_d'\boldsymbol{\Sigma}_{q(\theta_k)}\boldsymbol{x}_d\} + \sum_{n=1}^{N_d}\phi_{dnk}. \tag{A.1}$$

The variational parameters for topic assignment of word w_{dn} are obtained from

$$\begin{aligned}\boldsymbol{\phi}_{dn} \propto \boldsymbol{\tau}_n \exp\Big\{&\Psi(\boldsymbol{\zeta}_d) + \frac{1}{N_d}\frac{a_{q(\sigma^2)}}{b_{q(\sigma^2)}}\sum_{i\in\mathcal{I}_{s,d}}(y_{di} - \chi_{id}\mu_{q(\alpha)})\boldsymbol{\mu}_{q(\gamma_i)} \\ &- \frac{1}{2N_d^2}\frac{a_{q(\sigma^2)}}{b_{q(\sigma^2)}}\sum_{i\in\mathcal{I}_{s,d}}\big[2(\boldsymbol{\Sigma}_{q(\gamma_i)} + \boldsymbol{\mu}_{q(\gamma_i)}\boldsymbol{\mu}_{q(\gamma_i)}')\boldsymbol{\phi}_{d,-n} + \boldsymbol{\mu}_{q(\gamma_i)}^2 \\ &+ \mathrm{diag}(\boldsymbol{\Sigma}_{q(\gamma_i)})\big]\Big\},\end{aligned} \tag{A.2}$$

where $\boldsymbol{\phi}_{d,-n} = \sum_{j\neq n}\boldsymbol{\phi}_j$, and $\Psi(\cdot)$ is the digamma function. Let $\mathcal{I}_{s,d} = \mathcal{I}_d \cap \mathcal{S}^{(t)}$ denote the sampled individuals who rated document d.

We update the variational parameters of coefficient $\boldsymbol{\gamma}_i$ for every sampled user,

$$\boldsymbol{\Sigma}_{q(\gamma_i)}^{-1} = \rho_{q(\Gamma)}\boldsymbol{R}_{q(\Gamma)}^{-1} + \frac{a_{q(\sigma^2)}}{b_{q(\sigma^2)}}\mathrm{E}(A_i'A_i), \tag{A.3}$$

$$\boldsymbol{\mu}_{q(\gamma_i)} = \boldsymbol{\Sigma}_{q(\gamma_i)}\big[\rho_{q(\Gamma)}\boldsymbol{R}_{q(\Gamma)}^{-1}\boldsymbol{\mu}_{q(\beta)} + \frac{a_{q(\sigma^2)}}{b_{q(\sigma^2)}}\mathrm{E}(A_i')(\boldsymbol{y}_i - \mu_{q(\alpha)}\boldsymbol{\chi}_i)\big],$$

where $\mathrm{E}(A_i) = \{\bar{\boldsymbol{\phi}}_d\}_{d\in\mathcal{D}_i}$, and

$$\mathrm{E}(A_i'A_i) = \sum_{d\in\mathcal{D}_i}\frac{1}{N_d^2}\sum_{n=1}^{N_d}\Big[\sum_{m\neq n}\boldsymbol{\phi}_{dn}\boldsymbol{\phi}_{dm}' + \mathrm{diag}(\boldsymbol{\phi}_{dn})\Big]. \tag{A.4}$$

Next, we update the population parameters using the exponential smoothing and rescaling discussed in Appendix C. Specifically, the topic distribution is given by

$$\tau_{kw} \propto (1-\pi)\tau_{kw}^{(t-1)} + \pi\sum_{d=1}^{D}\sum_{n=1}^{N_d}1(w_{dn} = w)\phi_{dnk}. \tag{A.5}$$

The variational parameters for the population covariance matrix $\boldsymbol{\Gamma}$ are calculated as

$$\rho_{q(\Gamma)} = \rho_\Gamma + I, \tag{A.6}$$

$$\begin{aligned}\boldsymbol{R}_{q(\Gamma)} = (1-\pi)\boldsymbol{R}_{q(\Gamma)}^{(t-1)} + \pi\Big\{&\boldsymbol{R}_\Gamma + I\boldsymbol{\Sigma}_{q(\beta)} \\ &+ \frac{I}{|\mathcal{S}^{(t)}|}\sum_{i\in\mathcal{S}^{(t)}}[(\boldsymbol{\mu}_{q(\beta)} - \boldsymbol{\mu}_{q(\gamma_i)})(\boldsymbol{\mu}_{q(\beta)} - \boldsymbol{\mu}_{q(\gamma_i)})' + \boldsymbol{\Sigma}_{q(\gamma_i)}]\Big\}.\end{aligned}$$

The variational parameters for the population mean $\boldsymbol{\beta}$ are given by

$$\boldsymbol{\Sigma}_{q(\beta)}^{-1} = (1-\pi)\boldsymbol{\Sigma}_{q(\beta)}^{-1} + \pi(\boldsymbol{\Sigma}_\beta^{-1} + I\cdot\rho_{q(\Gamma)}\boldsymbol{R}_{q(\Gamma)}^{-1}), \tag{A.7}$$

$$\boldsymbol{\mu}_{q(\beta)} = (1-\pi)\boldsymbol{\mu}_{q(\beta)}^{(t-1)} + \pi\boldsymbol{\Sigma}_{q(\beta)}\Big(\boldsymbol{\Sigma}_\beta^{-1}\boldsymbol{\mu}_\beta + \rho_{q(\Gamma)}\boldsymbol{R}_{q(\Gamma)}^{-1}\frac{I}{|\mathcal{S}^{(t)}|}\sum_{i\in\mathcal{S}^{(t)}}\boldsymbol{\mu}_{q(\gamma_i)}\Big).$$

The variational parameters for the selection correction coefficient α are calculated as

$$\sigma_{q(\alpha)}^{-2} = (1-\pi)\sigma_{q(\alpha)}^{-2\,(t-1)} + \pi\Big[\sigma_\alpha^{-2} + \frac{a_{q(\sigma^2)}}{b_{q(\sigma^2)}}\frac{I}{|\mathcal{S}^{(t)}|}\sum_{i\in\mathcal{S}^{(t)}}\boldsymbol{\chi}_i'\boldsymbol{\chi}_i\Big], \tag{A.8}$$

$$\begin{aligned}\mu_{q(\alpha)} = &(1-\pi)\mu_{q(\alpha)}^{(t-1)} \\ &+ \pi\sigma_{q(\alpha)}^2\Big[\sigma_\alpha^{-2}\mu_\alpha + \frac{a_{q(\sigma^2)}}{b_{q(\sigma^2)}}\frac{I}{|\mathcal{S}^{(t)}|}\sum_{i\in\mathcal{S}^{(t)}}\boldsymbol{\chi}_i'(\boldsymbol{y}_i - \mathrm{E}(A_i)\boldsymbol{\mu}_{q(\gamma_i)})\Big].\end{aligned}$$

The variational parameters for the error variance σ^2 are given by

$$a_{q(\sigma^2)} = a_{\sigma^2} + \frac{1}{2}\sum_{d=1}^{D}|\mathcal{I}_d|, \tag{A.9}$$

$$\begin{aligned}b_{q(\sigma^2)} = (1-\pi)b_{q(\sigma^2)}^{(t-1)} + \pi\Big\{&b_{\sigma^2} + \frac{1}{2}\frac{I}{|\mathcal{S}^{(t)}|}\sum_{i\in\mathcal{S}^{(t)}}\Big(\|\boldsymbol{y}_i - \mathrm{E}(A_i)\boldsymbol{\mu}_{q(\gamma_i)} \\ &- \mu_{q(\alpha)}\boldsymbol{\chi}_i\|^2 + \mathrm{Tr}[\mathrm{E}(A_i'A_i)\boldsymbol{\Sigma}_{q(\gamma_i)}] + \sigma_{q(\alpha)}^2\|\boldsymbol{\chi}_i\|^2\Big)\Big\}.\end{aligned} \tag{A.10}$$

Finally, we apply the ADSVB procedure described in Appendix B to obtain $q(\boldsymbol{\theta}_k) = \mathcal{N}(\boldsymbol{\mu}_{q(\theta_k)}, \boldsymbol{\Sigma}_{q(\theta_k)})$, $k = 1\ldots K$. Because of the summation over the sampled documents, the gradient calculation in the ADSVB procedure is subject to the rescaling we discuss in Appendix C.

Appendix B. ADSVB Estimation

We now detail the adaptive doubly stochastic variational Bayesian updating for the nonconjugate model component $\boldsymbol{\theta}_k$.

Input: standard normal density $\phi(\boldsymbol{s})$ and gradient $\nabla \log p(\boldsymbol{\omega}, \boldsymbol{\theta}_k)$
Setup: step size ϖ, constants ϵ and ε, and exponential decay rates $\kappa_1, \kappa_2 \in [0, 1)$
Initialize: $\boldsymbol{\mu}_{q(\theta_k)}$, $\boldsymbol{C}$, t, $\boldsymbol{m}_\mu$, $\boldsymbol{v}_\mu$, $\boldsymbol{m}_C$, and v_C
Repeat till convergence:

$t = t + 1$
$\boldsymbol{s} \sim \phi(\boldsymbol{s})$
$\boldsymbol{\theta}_k = \boldsymbol{C}\boldsymbol{s} + \boldsymbol{\mu}_{q(\theta_k)}$
$\kappa_{1t} = \kappa_1 \cdot \varepsilon^{t-1}$
For $\boldsymbol{\mu}_{q(\theta_k)}$:

$\boldsymbol{g}_\mu = \nabla \log p(\boldsymbol{\omega}, \boldsymbol{\theta}_k)$ (Get gradient for $\boldsymbol{\mu}$)
$\boldsymbol{m}_\mu = \kappa_{1t} \cdot \boldsymbol{m}_\mu + (1 - \kappa_{1t}) \cdot \boldsymbol{g}_\mu$ (Update biased 1st moment)
$\boldsymbol{v}_\mu = \kappa_2 \cdot \boldsymbol{v}_\mu + (1 - \kappa_2) \cdot \boldsymbol{g}_\mu^2$ (Update biased 2nd moment)
$\widehat{\boldsymbol{m}}_\mu = \boldsymbol{m}_\mu / (1 - \kappa_1^t)$ (Correct bias for 1st moment)
$\widehat{\boldsymbol{v}}_\mu = \boldsymbol{v}_\mu / (1 - \kappa_2^t)$ (Correct bias for 2nd moment)
$\boldsymbol{\mu}_{q(\theta_k)} = \boldsymbol{\mu}_{q(\theta_k)} - \varpi \cdot \widehat{\boldsymbol{m}}_\mu / (\sqrt{\widehat{\boldsymbol{v}}_\mu} + \epsilon)$ (Update variational parameter)

For $\boldsymbol{\Sigma}_{q(\theta_k)}$:

$\boldsymbol{g}_C = \nabla \log p(\boldsymbol{\omega}, \boldsymbol{\theta}_k) \times \boldsymbol{s}' + \triangle_C$
$\boldsymbol{m}_C = \kappa_{1t} \cdot \boldsymbol{m}_C + (1 - \kappa_{1t}) \cdot \boldsymbol{g}_C$
$\boldsymbol{v}_C = \kappa_2 \cdot \boldsymbol{v}_C + (1 - \kappa_2) \cdot (\boldsymbol{g}_C \dot{\circ} \boldsymbol{g}_C)$
$\widehat{\boldsymbol{m}}_C = \boldsymbol{m}_C / (1 - \kappa_1^t)$
$\widehat{\boldsymbol{v}}_C = \boldsymbol{v}_C / (1 - \kappa_2^t)$
$\boldsymbol{C} = \boldsymbol{C} - \varpi \cdot \widehat{\boldsymbol{m}}_C / (\sqrt{\widehat{\boldsymbol{v}}_C} + \epsilon)$
$\boldsymbol{\Sigma}_{q(\theta_k)} = \boldsymbol{C}\boldsymbol{C}'$

In the algorithm, $\triangle_C$ denotes the diagonal matrix with $\{1/C_{11}, \ldots, 1/C_{mm}\}$ on its diagonal. The gradient of the log-joint density is derived as

$$\frac{\partial \log p(\boldsymbol{\omega}, \boldsymbol{\theta}_k)}{\partial \boldsymbol{\theta}_k} = \sum_{d=1}^{D} u_{dk} \Big[\Psi\Big(\sum_{j=1}^{K} u_{dj} \Big) - \Psi(u_{dk}) + \Psi(\zeta_{dk}) - \Psi\Big(\sum_{j=1}^{K} \zeta_{dj} \Big) \Big] \boldsymbol{x}_d - \boldsymbol{\Sigma}_{\theta_k}^{-1} (\boldsymbol{\theta}_k - \boldsymbol{\mu}_{\theta_k}), \tag{B.1}$$

where $u_{dk} = \exp(\boldsymbol{x}_d' \boldsymbol{\theta}_k)$. Additional technical details are available from the authors upon request.

Appendix C. SVB Estimation with Adaptive Minibatch Sizes

To adaptively adjust the minibatch size to achieve faster convergence, we start the SVB estimation by sampling small batches of the data, and increase the batch size in a VB iteration only if information from the current batch no longer suffices to move the optimization toward the appropriate direction. We follow the RPP strategy in Tan (2015) to determine whether to keep or to change batch size in an iteration. The RPP indicates the extent to which the parameter values are moving monotonically toward optimum in the past M iterations, as opposed to merely bouncing back and forth around the optimum.

Denote by ν the generic representation of the population parameter of interest. In iteration t, the RPP is calculated as (Gaivoronski 1988)

$$\text{RPP} = \frac{|\nu^{(t-M)} - \nu^{(t)}|}{\sum_{r=t-M}^{t-1} |\nu^{(r)} - \nu^{(r+1)}|}, \tag{C.1}$$

where $\nu^{(t)}$ is the scalar component of the population parameter vector in iteration t. The RPP is bounded between zero and one. It equals zero if $\nu^{(l-M)} = \nu^{(l)}$, a sign that no real progress was made after M iterations. It equals one if the path from $\nu^{(l-M)}$ to $\nu^{(l)}$ is perfectly monotonic. Between zero and one, a small RPP implies that the optimization process exhibits considerable oscillating behavior, and thus more resolution and information are needed from data, whereas a big RPP signals relatively more monotonic progress has been made. In iteration t, if the RPP falls below a prespecified threshold ϱ, we increase the batch size by a fixed percentage. When the minibatch reaches the full data set, we continue the optimization by switching to the nonstochastic VB method till convergence. Using the RPP indicator, we essentially allow the algorithm to determine by itself the most appropriate batch size to have during the optimization process.

In applying adaptive minibatch sizes to the main model, we take a random sample $\mathcal{S}^{(t)}$ from the I individuals in iteration t and calculate RPP at the end of the iteration, until the minibatch reaches the full data. For each sampled individual, the computation of their variational parameters remains the same as before. The updating for the population parameters now includes an exponential smoothing of the variational results between the current and the last iterations with weight π. Such weighted average ensures the convergence of the stochastic optimization (Hoffman et al. 2013). Also, any terms that involve summation over the sampled individual parameters are rescaled by a multiplier $I/|\mathcal{S}^{(t)}|$, as if the entire data set (i.e., all individuals) was used for the current updating. The nonstochastic version of VB algorithm is recovered when $\pi = 1$ and $|\mathcal{S}^{(t)}| = I$, that is, the minibatch includes all individuals from the original data set. For faster convergence, we allow higher tolerance and less precision in early iterations by changing the RPP threshold ϱ and the weight π linearly with the batch size across iterations.

Endnotes

[1] Alternative stochastic strategies include taking samples on documents (movies) rather than on individuals in each iteration, or sampling both documents and users in the same iteration. In the MovieLens data, there are many more individuals than movies, and thus sampling users is more beneficial.

[2] For a nonlinear model, prediction based on posterior parameter means can be different from prediction based on averaging posterior predictive samples. In our case, however, the predicted ratings are obtained via a linear combination of the empirical frequencies of different topics in the product descriptions, that is, the rating equation is linear in γ_i conditional on other unknowns, and therefore Jensen's inequality has little impact on predicting the ratings. Also, given the trade-off between scalability and accuracy in the recommendation context, our prediction based on point estimates is sufficiently accurate given the goal of providing timely estimates.

[3] The insignificant impact of selectivity correction could be due to multiple reasons. For instance, selection bias could be minimal, or the data (also from MovieLens) used to model the first stage could be unable to tease out the selection sources. The setups of searching, filtering, ordering, and displaying movies could separately or jointly affect whether a movie is rated, and there are numerous factors external to MovieLens, such as promotions and word of mouth, that may affect a user's awareness of a movie. As we have no information about these aspects beyond what is available in the MovieLens database, modeling the exact consideration set is nearly impossible.

[4] For a fair comparison we coded both the SVB and MCMC methods using Mathematica 11.1 and used its just-in-time compilation capability to compile the programs to C. We ran all programs on a computer with a 3 GHz 8-Core Intel Xeon E5 processor and 32 GB of RAM.

[5] We simulated 10,000 draws from $q(\gamma_i)$ of a user to compute the rank probabilities.

References

Ansari A, Essegaier S, Kohli R (2000) Internet recommendation systems. *J. Marketing Res.* 37(3):363–375.

Bishop C (2006) *Pattern Recognition and Machine Learning* (Springer, New York).

Blei D, McAuliffe J (2007) Supervised topic models. *Proc. 20th Annual Internat. Conf. Advances in Neural Information Processing Systems* (Curran Associates, Red Hook, NY), 121–128.

Blei D, Ng A, Jordan M (2003) Latent Dirichlet allocation. *J. Machine Learn. Res.* 3(January):993–1022.

Bodapati AV (2008) Recommendation systems with purchase data. *J. Marketing Res.* 45(1):77–93.

Braun M, McAuliffe J (2010) Variational inference for large-scale models of discrete choice. *J. Amer. Statist. Assoc.* 105(489):324–335.

Büschken J, Allenby GM (2016) Sentence-based text analysis for customer reviews. *Marketing Sci.* 35(6):953–975.

Chung J, Rao VR (2012) A general consumer preference model for experience products: Application to internet recommendation services. *J. Marketing Res.* 49(3):289–305.

Chung TS, Rust RT, Wedel M (2009) My mobile music: An adaptive personalization system for digital audio players. *Marketing Sci.* 28(1):52–68.

de Gemmis M, Lops P, Semeraro G, Basile P (2008) Integrating tags in a semantic content-based recommender. *Proc. 2008 ACM Conf. Recommender Systems* (ACM, New York), 163–170.

Desrosiers C, Karypis G (2011) A comprehensive survey of neighborhood-based recommendation methods. Ricci F, Rokach L, Shapira B, Kantor P, eds. *Recommender Systems Handbook* (Springer, Boston), 107–144.

Dzyabura D, Hauser JR (2011) Active machine learning for consideration heuristics. *Marketing Sci.* 30(5):801–819.

Erosheva E, Fienberg S, Lafferty J (2004) Mixed-membership models of scientific publications. *Proc. Natl. Acad. Sci. USA* 101(Suppl 1): 5220–5227.

Firan CS, Nejdl W, Paiu R (2007) The benefit of using tag-based profiles. *Proc. 2007 Latin American Web Conf.* (IEEE Computer Society, Washington, DC), 32–41.

Gaivoronski A (1988) Implementation of stochastic quasigradient methods. Ermoliev Y, and R.J.B. Wets RJB, eds. *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, New York), 313–352.

Giordano R, Broderick T, Jordan MI (2017) Covariances, robustness, and variational Bayes. Working paper, University of California, Berkeley, Berkeley.

Greene WH (2003) *Econometric Analysis*, 5th ed. (Pearson Education, New York).

Hall P, Ormerod JT, Wand MP (2011) Theory of Gaussian variational approximation for a Poisson mixed model. *Statistica Sinica* 21(1): 369–389.

Harper M, Konstan JA (2015) The MovieLens datasets: History and context. *ACM Trans. Interactive Intelligent Systems* 5(4):1–19.

Heckman J (1979) Sample selection bias as a specification error. *Econometrica* 47(1):153–161.

Hoffman M, Blei DM, Wang C, Paisley J (2013) Stochastic variational inference. *J. Machine Learn. Res.* 14(1):1303–1347.

Jin X, Zhou Y, Mobasher B (2005) A maximum entropy web recommendation system: Combining collaborative and content features. *Proc. 11th ACM SIGKDD Internat. Conf. Knowledge Discovery in Data Mining* (ACM, New York), 612–617.

Jordan MI, Ghahramani Z, Jaakkola TS, Saul LK (1999) An introduction to variational methods for graphical models. *Machine Learn.* 37(2):183–233.

Keynes JM (1921) A treatise on probability. *Collected Writings of John Maynard Keynes*, Vol. 8 (Macmillan, London).

Knowles DA, Minka TP (2011) Non-conjugate variational message passing for multinomial and binary regression. *Proc. 24th Annual Internat. Conf. Advances in Neural Information Processing Systems* (Curran Associates, Red Hook, NY), 1701–1709.

Koren Y, Bell R (2011) Advances in collaborative filtering. Ricci F, Rokach L, Shapira B, Kantor P, eds. *Recommender Systems Handbook* (Springer, Boston), 145–186.

Kullback S, Leibler RA (1951) On information and sufficiency. *Ann. Math. Statist.* 22(1):79–86.

Lops P, Gemmis M, Semeraro G (2011) Content-based recommender systems: State of the art and trends. Ricci F, Rokach L, Shapira B, Kantor P, eds. *Recommender Systems Handbook* (Springer, Boston), 73–105.

Luts J, Ormerod JT (2014) Mean field variational Bayesian inference for support vector machine classification. *Comput. Statist. Data Anal.* 73:163–176.

Michlmayr E, Cayzer S (2007) Learning user profiles from tagging data and leveraging them for personalized information access. *Proc. 16th Internat. World Wide Web Conf. Tagging and Metadata for Social Inform. Organ., Banff, Canada*, 1–7.

Nam H, Kannan PK (2014) The informational value of social tagging networks. *J. Marketing* 78(4):21–40.

Nikulin MS (2001) Hellinger distance. Hazewinkel M, ed. *Encyclopedia of Mathematics* (Springer Science+Business Media B.V./Kluwer Academic Publishers, Berlin).

Ormerod JT, Wand MP (2010) Explaining variational approximations. *Amer. Statist.* 64(2):140–153.

Rezende DJ, Mohamed S (2016) Variational inference with normalizing flows. Working paper, Google, London.

Robbins H, Monro S (1951) A stochastic approximation method. *Ann. Math. Statist.* 22(3):400–407.

Rossi PE, McCulloch RE, Allenby GM (1996) The value of purchase history data in target marketing. *Marketing Sci.* 15(4):321–340.

Salakhutdinov R, Mnih A (2008) Bayesian probabilistic matrix factorization using Markov chain Monte Carlo. *Proc. 25th Internat. Conf. Machine Learn.* (ACM, New York), 880–887.

Salimans T, Knowles DA (2013) Fixed-form variational posterior approximation through stochastic linear regression. *Bayesian Anal.* 8(4):837–882.

Spall J (2003) *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control* (John Wiley & Sons, Hoboken, NJ).

Sudhir K (2016) Editorial: The exploration-exploitation tradeoff and efficiency in knowledge production. *Marketing Sci.* 35(1):1–9.

Szomszor M, Cattuto C, Alani H, O'Hara K, Baldassarri A, Loreto A, Servedio VDP (2009) Folksonomies, the semantic web, and movie recommendation. *Proc. Workshop Bridging Gap Semantic Web and Web 2.0, Innsbruck, Austria*.

Tan LSL (2015) Stochastic variational inference for large-scale discrete choice models using adaptive batch sizes. *Statist. Comput.* 27(1): 237–257.

Tirunillai S, Tellis GJ (2014) Mining marketing meaning from online chatter: Strategic brand analysis of big data using latent Dirichlet allocation. *J. Marketing Res.* 51(4):463–479.

Titsias MK, Lazaro-Gredilla M (2014) Doubly stochastic variational Bayes for non-conjugate inference. *Proc. 31st Internat. Conf. Machine Learn.* (ACM, New York), 1971–1979.

Titterington DM, Wang B (2006) Convergence properties of a general algorithm for calculating variational Bayesian estimates for a normal mixture model. *Bayesian Anal.* 1(3):625–650.

Wallach H, Mimno DM, McCallum A (2009) Rethinking LDA: Why priors matter. *Proc. 22nd Annual Internat. Conf. Advances*

in Neural Information Processing Systems (Curran Associates, Red Hook, NY), 1973–1981.

Wang C, Blei DM (2013) Variational inference in nonconjugate models. *J. Machine Learn. Res.* 14(1):1005–1031.

Wedel M, Kannan PK (2016) Marketing analytics for data-rich environments. *J. Marketing* 80(6):97–121.

Ying Y, Feinberg F, Wedel M (2006) Leveraging missing ratings to improve online recommendation systems. *J. Marketing Res.* 43(3): 355–365.

You C, Ormerod JT, Müller S (2014) On variational Bayes estimation and variational information criteria for linear regression models. *Australian New Zealand J. Statist.* 56(1):73–87.