Modeling Users’ Adoption Behaviors with Social Selection and Influence*

Ziqi Liu† Fei Wang‡ Qinghua Zheng†

Abstract

Massive users’ online adoption behaviors were recorded thanks to the various emerging web services such as Facebook, Twitter, G+ and Netflix and so on. Two key factors that affect users’ adoption behaviors are social selection and social influence. Understanding such factors underlying each behavior can potentially help web service providers gain much more insights into their users and improve predictive power. In this paper, we try to answer (1) How do the roles of selection and influence play in a user-level adoption? (2) Capturing those factors can benefit the modeling and prediction of users’ adoption behaviors or not. Quantitatively capturing the two factors could be challenging since the known “ballot box communication”. Moreover, though both social selection and influence are well studied in collaborative filtering and information diffusions respectively, it’s still non-trivial to jointly model them. We propose a probabilistic Latent Factors with Diffusion Model (LFDM) which explicitly considers both social selection and influence by projecting cascading processes into latent factor spaces. We also develop an effective EM styled algorithm for estimating the proposed model. Finally we validate our methodology on three kinds of real world data sets.

1 Introduction

With the popularity of various social environments, massive users’ online adoption behaviors were recorded. For example, users follow others on Twitter; customers buy items on Amazon; users review products on Epinions. Understanding users’ adoption behaviors is of crucial importance as this could be helpful in many applications such as item recommendation and advertisement targeting. However, this problem is challenging because users in social network encounter a huge space of items and there could be different factors that affect their decisions. Two key factors that affect users’ adoption behaviors are social selection and social influence, which are in tension and their interplay forms the everyday social processes [9]. Social selection refers to the phenomenon that users tend to form new links to others who already liked them. Social influence occurs when someone’s emotions, opinions, or behaviors are affected by his or her friends.

Computational tools have been developed to explore those factors. On one hand, social selection forms the basis of collaborative filtering (CF) [15, 16], which has proven to be a powerful method for identifying each user’s tastes by discovering the latent groups of people who have similar tastes. On the other hand, social influence plays a central role in influence diffusion [24], viral marketing [17], where influential actors could influence on others’ decisions. Most of the existing works focus on either influence or selection. However, it is important to consider them both to better understand the underlying user behaviors.

The benefits of modeling the two social factors for analyzing users’ behaviors are two-fold:

- It is important to discriminate which factor leads to a specific adoption. Such explanatory analysis could give online social network service providers more insights to make their service more popular. For instance, if a user’s major adoptions are caused by selection, it implies few social interactions exist, then the online services should focus on leading the user to be aware of his/her potential friends; if a user’s major adoptions are caused by social influence from his/her neighbors, that implies his/her potential personal interests could be drowned because of the conformity [29] with his/her neighbors, this suggests that the online services could focus on exploring the user’s existing interests to make him/her be aware of related items.

- Accurately capturing such social factors could benefit the prediction of users’ future behaviors.

However, it is difficult to jointly analyze the two social factors and the role they played in users’ adoption behaviors. On one hand, in most of online services, only friendship and item adoptions over time can be observed, but we do not know who affects whom, i.e. the two factors are latent. This is known as “ballot box
communication” [13]. It is non-trivial to infer the cause of each adoption from those observations. Second, social selection and influence were modeled from very different perspectives in the past. For example, collaborative filtering methods decompose users’ behaviors into users’ interests and characteristics of items [15]. Whereas, influence models focus on the diffusion patterns of each item [24]. It is still unknown how to jointly consider these two factors in a unified model.

In this paper, we focus on understanding the roles of selection and influence play in a user-level adoption, and studying whether capturing those factors can benefit the modeling and prediction of users’ adoption behaviors. Intuitively, we naturally incorporate latent interest spaces and latent influence spaces into a probabilistic framework, so that we can study the social selection and influence in a common foot. We further have the following considerations based on the framework:

- **Preferred Items.** We derive user $i$’s preferred item factors by projecting $i$’s potential influential behaviors upon the global latent item factors which are defined in latent interest space. Therefore, we can transform the exposure function learned from a diffusion model into our framework, thus naturally model the cascading series.

- **Love of the Same** [20]. Users’ tastes can be indicated as a mixture of both users with similar tastes (selection) and existing friends’ tastes (influence). Carefully modeling users’ interests can combat sparsity and most importantly result in an unbiased estimation of the cause for each adoption.

To summarize, this is the first work combines the two social factors, i.e. selection and influence in a unified model. We name it Latent Factors with Diffusion Model (LFDM). We effectively estimate our model in an EM styled algorithm by alternatively inferring the latent variables and estimating the parameters. Finally, we evaluate our method by ranking measures, and illustrate the tendency to be influenced in a user-level using three real-world data.

2 Related Work

Information Diffusion. Social influence founds the basis of information diffusion or propagation. Works [10, 13, 19, 21, 24] in this domain assumed that each user is either active (infected, influenced) or inactive, and only active users can spread the contagion (information, disease) along the friendship network. Careful targeting can make a cascading effect on the adoption behaviors. Those works are mainly interested in (a) where information should diffuse; (b) how much time it diffuses; (c) how the expected cascading scales. Both discrete and continuous [10, 21] diffusion models were studied in this scenario. However, they did not intend to distinguish or capture the users’ own tastes. Moreover, those works have no predictive power on unexplored items which are not adopted by one’s circles.

Collaborative Filtering (CF). Social selection forms the basis of CF where similar historic user behaviors imply similar tastes. Both neighbor-based models and latent factor models have been proved to be powerful in analyzing adoptions via similarity ties. A good comparative study can be found in [15]. Moreover, to incorporate social network context, social recommendation was proposed. They optimized the latent factor models by regularizing the latent user factors through social ties [18, 30], or constraining latent user factors through another group space [20, 31]. Even though such models strengthen their predictive power by exploiting the correlation among friends, they do not analyze and distinguish the causes of users’ adoptions. In addition, [22] combines the features of both CF and diffusion models. However, they propose a specific method only for social update recommendations, and don’t intend to capture the two social factors either.

Social Selection and Influence. It is important to understand the relative effects of social selection and influence on social adoption behaviors. The works [2, 9] in this domain mainly qualitatively measure the interplay between selection and influence: (a) they focus on analyzing the effects of social influence by simply define one user’s historic activities [9] or user profiles [3] as indicators of social selection; (b) or they [2] try to test whether the social factors exist in a social network; (c) the items are considered independent of each other. However, there is no existing method which can unify the two social factors in one model.

3 Methods

3.1 Problem Settings Assuming a social network with $|U|$ users $u = \{1,...,i,...,|U|\}$ and $|R|$ relationships $r$. The relationship between any two users $i$ and $i'$ could be either directed (e.g. trust network) or undirected (e.g. friend network). There are $|V|$ potential items $v = \{1,...,j,...,|V|\}$ can be adopted. We are also given a series of observed adoptions $a = \{a_1,...,a_t,...,a_T\}$, where each adoption $a_t$ can be overloaded as $a_t^{i,j}$ which means user $i$ adopts item $j$ at time $t$. We want to explore the cause (i.e. social influence or selection) for each adoption $a_t^{i,j}$. If one’s adoption is caused by social selection, that implies the adoption depends on the tastes of himself/herself; if the adoption is caused by social influence, that implies the adoption is triggered by behaviors of his or her friends. Moreover,
by characterizing the role social selection and influence played on each user’s adoptions, we aim at predicting users’ behaviors. That is, we want to build a model with both explanatory ability and predictive power.

Social selection can be modeled by collaborative filtering (CF) methods. In this paper, we only focus on latent factor models

\[ a_{i,j} \sim P(a_{i,j} | \theta_i^T \cdot \phi_j) \]

where we associate latent factors \( \theta_i \in \mathbb{R}^K \) with each user \( i \) and \( \phi_j \in \mathbb{R}^K \) with each item \( j \). Respectively, they represent user \( i \)’s preferences over a latent interest space, and item \( j \)’s characteristics over the same latent space. Let \( K \) denote the number of latent factors (interests).

Conventionally there are two kinds of methodologies work well in this framework. First, we can do regression over the latent factors \( \theta_i \) and \( \phi_j \). One can do penalized MLE or minimize various loss functions [30] such as logistic regression, Huber loss and so on. Second, we can view it from a generative perspective where parameters themselves are probability distributions and satisfy the sum-to-one constraints. One popular treatment is to impose a Dirichlet prior [5], since the natural property where each sample from a Dirichlet distribution is itself a probability distribution. Comparatively speaking, the former methodology has some benefits such as an easy way of deriving a scalable algorithm, various kinds of loss functions. However the drawbacks are also obvious. (1) Lack of interpretability. It means the learned parameters \( \theta_i \) and \( \phi_j \) cannot be interpreted intuitively. (2) It is nontrivial to handle the dyadic responses when only positive data are observed. One has to consider pseudo-negative samples [30] or sweep all the entries [12] in the optimization procedure which offsets the advantage of computational cost. In this paper, we consider the above framework in a generative manner because (1) dyadic responses with only positive instances can be naturally modeled, (2) we stress the interpretability of the adoption process, (3) we intend to uncover the latent structures underlying the process in a principled Bayesian way.

For information diffusion, researchers usually make the independence assumption between different items. One purpose is to learn an exposure function [2, 21]. To make the formulation consistent with those works, we define the exposure function by Eq. (3.2), where \( I_{t-\delta}(i,j) \) indicates the number of times user \( i \) adopts item \( j \) at time \( t-\delta \), and \( \psi_{f_i} \) represents the influential strength from \( f \) to \( i \) with property \( \sum_f \psi_{f_i} = 1 \), and \( \nu \) is the decayed parameter. Various decayed functions were studied [10]. We consider exponential functions for simplicity. Eq. (3.2) means whether user \( i \) adopted item \( j \) at time \( t \) depends both on the time user \( i \)’s

\[
a_{i,j}^t \sim P(a_{i,j}^t | \sum_{f \in \text{friends}} \sum_{\delta=1}^{\Delta} \exp^{-\delta/\nu} \cdot \psi_{f_i} \cdot I_{t-\delta}(f,j))
\]

We summarize the notations that will be used throughout this paper in Table [1]

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( i )</td>
<td>user index</td>
<td>( j )</td>
<td>user index</td>
</tr>
<tr>
<td>( a_{i,j}^t )</td>
<td>( i ) adopts ( j ) at time ( t )</td>
<td>( \Psi )</td>
<td>user-user influence matrix</td>
</tr>
<tr>
<td>( \Theta )</td>
<td>user-interest matrix</td>
<td>( \Phi )</td>
<td>interest-item matrix</td>
</tr>
<tr>
<td>( \psi_{f,j,i} )</td>
<td>one entry of ( \psi_t )</td>
<td>( \psi_k )</td>
<td>the influence from others to ( i )</td>
</tr>
<tr>
<td>( \theta_{i,k} )</td>
<td>one entry of ( \theta_t )</td>
<td>( \Theta_{i} )</td>
<td>( i )’s interest distribution</td>
</tr>
<tr>
<td>( \phi_k )</td>
<td>popularity over items given interest ( k )</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \phi_{j,k} )</td>
<td>characteristics of ( j ) over interest space</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \phi_{i,t} )</td>
<td>( i )’s preferred interest-item matrix at ( t )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

3.2 Models We first propose a framework which combines latent influence spaces with latent interest spaces, then develop our Latent Factors with Diffusion Model (LFDM) step by step.

We introduce a new parameter \( \Psi \in \mathbb{R}^{|U| \times |U|} \), where each entry \( \psi_{i',i} \) represents \( i \)’s influence on \( i' \) and we have \( \sum_{i \in U} \psi_{i,i'} = 1 \). In practice the matrix \( \Psi \) is sparse (\( \psi_{i,i'} > 0 \) iff \( i \) and \( i' \) are friends) and asymmetric. The diagonal entries represent the influence strength on oneself. It can be viewed as the probability of one’s adoption that is caused by social selection. Combining with latent factor models depicted in Eq. (3.1), we get the following framework:

\[ a_{i,j} \sim P(a_{i,j} | \psi_{i,i}^T \cdot \Theta \cdot \phi_j) \]

Some works [7, 32] have employed this framework to model social correlation. However, they essentially do not consider social influence. To accurately model social influence and selection in this framework, we have the following considerations:

(1) Love of the Same [20]. Whenever one makes an adoption, no matter it appears to indicate a brand new interest developed by the user or not, we have no idea that the interest is exactly new due to influence
or the interest already exists in one’s mind but did not show before especially in sparse data. What we know is each user’s interests could be indicated by both those users with similar tastes and those users already connected. Both of the two social factors would help the exploration of users’ interests. One should place a reasonable prior for such structure and distinguish the two social factors based on observations. However, related works [7] [22] do not explicitly capture such structure, hence could potentially bias one adoption as a result of social influence if the adoptor did not adopt any item of this kind before.

(2) Preferred Item. The parameter \( \phi_k \) can be viewed as a global variable which represents the global preference from all the users over the items given interest \( k \). This is indeed a reasonable model while capturing social selection. For social influence, once a user \( i \) became an adopter of item \( j \), there has a chance \( i \)’s neighbour \( i' \) can be infected later in the timeline, i.e. the cascading processes. Therefore, in addition to the global \( \phi_k \), we suppose each user \( i \) has a specific preference \( \phi_{ik} \) over \( i \)'s historic adopted items given interest \( k \) at time \( t \). We learn \( i \)'s preferred item factors \( \Phi_{i,t} \) at time \( t \) as a function of intensity functions, and substitute the preferred item factors for global item factors \( \Phi \) in Eq. (3.3), then we get the resultant model as an exposure function Eq. (3.2) considered in latent interest spaces. Note that even cascading series can be viewed as either continuous time series [13] or discrete ordinal time series [2] theoretically, in this paper, we consider ordinal time series for simplicity.

According to the former observations, we now specify our model based on the framework Eq. (3.3).

First, we model the structure love of the same underlaid \( \Theta \). We begin the modeling of \( \Theta \) by a hierarchical Polya-Urn representation [4]. Suppose that user \( i \) has generated \( j \) – 1 actions, let \( n_{ik} \) represent the number of actions of \( i \) expressing interest \( k \). To generate action \( j \) the user might choose to either repeat an old interest with probability proportional to \( n_{ik} \) or consider a brand new interest with probability proportional to \( \eta \). If a brand new interest is selected, user \( i \) can select to express interest \( k \) with probability proportional to \( m_k + \alpha / K \), where \( m_k \) represents the global frequency by which interest \( k \) is expressed across users. We introduce a latent variable \( z_{ij} \) for the corresponding adoption \( a_{ij} \), which indicates the probability \( a_{ij} \) is generated by interest \( z_{ij} \). Integrating out the interests \( \theta_i \), we have:

\[
P(z_{i,j} = k | \text{rest}) \propto n_{ik}^{-(i,j)} + \eta (m_{ik}^{-(i,j)} + \alpha / K) \sum_{k'} m_{k'}^{-(i,j)} + \alpha = Z_{ij}^k
\]

The superscript \( -(i,j) \) in those variables denotes the same count but without the adoption \( a_{ij} \). However, such representation cannot capture the potential similar interests between friends. So we further derive:

\[
P(z_{i,j} = k | z_{i,j}^{-1}) \propto m_{ik}^{-(i,j)} + \alpha / K \sum_{k'} m_{k'}^{-(i,j)} + \alpha = Z_{ij}^k
\]

where \( m_{ik}^{r,i} \) denotes the number of adoptions which are associated with interest \( k \) among user \( i \)'s influentials. The key point here is we don’t smooth over users’ latent factors by weighting their influence strength like other methods, we place structured priors over each user’s interests and learn it from data, thus we expect to avoid the potential bias for the cause of each adoption.

Second, we consider the modeling of \( \Phi \). We explicitly introduce the latent variable \( f_{i,j} \) which indicates the probabilities whom affects the adoption \( a_{ij} \). When the adoption \( a_{ij} \) is caused by selection, i.e. \( f_{i,j} = i \), we use the global parameter \( \Phi \) the same as in any latent factor model. When the adoption \( a_{ij} \) is influenced by a friend, i.e. \( f_{i,j} = f \), friend \( f \)'s preferred item factors \( \Phi_{f,t} \) at time \( t \) is computed instead:

\[
\phi_k^{f,t} | \Phi \propto \phi_k \sum_{\delta=1}^{\Delta} \exp^{-\delta/\tau} \cdot I_{t-\delta}(f,j)
\]

The key point is that instead of computing global \( \Phi \), we compute \( \Phi_{f,t} \), which measures user \( f \)'s personal preference over his or her adopted items by time \( t \), for adoptions which are influenced by friend \( f \). To calculate each \( \phi_k^{f,t} \), we first compute the intensity \( \lambda(f,j,t) \) which reflects the strength that \( f \) prefers \( j \) at time \( t \). We then project this intensity upon the interest spaces by reweighing the intensity via \( \phi_k \). \( \lambda(f,j,t) \). To make it a proper distribution, we normalize each vector \( \phi_k^{f,t} \). Note that \( \Phi_{f,t} \) is usually very sparse. The non-zero coordinates only take when the friend \( f \) has adopted item \( j \) and \( j \) was indeed interested in \( k \) before time \( t \).

Third, we expect that our model smooths in an evolutionary manner. Intuitively, as new friends are made, the influential strength \( \Psi \) should evolve. Each user’s interests \( \theta_i \) will be explored as new interests can be infected later in the timeline. Specifically, we model the evolution of \( \Psi \) and \( \Phi \) using a Gaussian random walk kernel. For the user-interest \( \Theta \) parameter, a similar
method with \cite{1} is introduced, where interests at time \( t \) depend on previous historic behaviors. Note that it is not required that the above variables evolve exactly the same granularity as Eq. (3.5), due to the reality that adoption data are sparse at each discrete time point.

To summarize, we assume each adoption \( a_{i,j}^t \) is caused by either \( i \)'s own decisions, or potential influencers. We decompose the behavior via \( \psi_i \cdot \theta_i^T \cdot \phi_j \) conditioned on social selection, or \( \psi_{ij} \cdot \theta_i^T \cdot \phi_{ij}^T \) conditioned on the social influence from \( f \). If there are no cascading effects, our model degenerates to a latent factor model. Instead of directly modeling intensities by a Poisson process, we project the intensities upon the latent interest spaces. The benefit is two-fold. (a) We can incorporate diffusion processes into the unified framework Eq. (3.3); (b) Social influence can be viewed from different angles \cite{28}. For example, the colleagues have strong influence on each other’s works, but may not on daily life. Our model naturally constrains the influence in a latent interest space. The more attention one shows on a given topic, the higher possibility he or she can affect others on the same topic. Our model smooths in an evolutionary manner, and can be formalized as:

\[
(3.6) \quad P(a, \Psi, \Theta, \Phi, f, z|\mathcal{H}) = \prod_{i=1}^T P(\Phi_i^t|\Phi_i^{t-1}, \mathcal{H}) \\
= \prod_i P(\Psi_i^t|\Psi_i^{t-1}, \mathcal{H})P(\theta_i^t|\theta_i^{t-1}, \mathcal{H}) \\
\times \prod_j P(f_{i,j}^t|\psi_i^t)P(z_{i,j}^t|f_{i,j}^t, \Theta^t)P(a_{i,j}^t|f_{i,j}^t, z_{i,j}^t, \Phi^t, \Phi_{i,j}^t, \Theta^t)
\]

where \( \mathcal{H} = \{a, \eta, \alpha\} \). To predict unknown data, we just compute the probability in Eq. (3.3).

4 Estimation

What we want to maximize the marginal probability:

\[
\text{max} \ P(a|\mathcal{H}) = \int_\Theta \int_\Phi \int_f \sum_z \sum_a P(a, \Psi, \Theta, \Phi, f, z|\mathcal{H})
\]

The parameters \( \Psi \) and \( \Theta \) are local variables which are associated with each user, and \( \Phi \) is a global variable shared across all the users.

E-step. The two latent indicators \( f, z \) are tightly coupled together. So we sample the latent variables \( f, z \) by Block-Gibbs sampling as follows:

\[
(4.7) \quad P(f_{i,j}^t = f, z_{i,j}^t = k|\text{rest}) \propto P(f_{i,j}^t = f|\psi_i^t)P(z_{i,j}^t = k|\alpha, \eta, z_{i,j}^{t-1}, f_{i,j}^{t-1}, f_{i,j}^t = f) \\
P(a_{i,j}^t|\phi_{kj}^t, \phi_{ij}^t, z_{i,j} = k, f_{i,j} = f)
\]

We expand for each term above in turn. For the prior over latent friend indicators, \( f_{i,j} \)'s are conditionally independent given the parameter \( \psi_i \). For the prior over latent topic indicator \( z \), we integrate out the \( \Theta_i \) for each user \( i \) independently. Thus, it forms an extension of the hierarchical Polya-Urn representation which has been discussed in Eq. (3.4). Note that \( z_{i,j} \) is also dependent on the latent friend indicator \( f_{i,j} \):

\[
(4.8) \quad P(z_{i,j}^t = k|\alpha, \eta, z_{i,j}^{t-1}, f_{i,j}^t = f) \propto u_{f_{i,j}}^t \eta_{f_{i,j}}^t + \eta_1 m_{k}^{t-1} + \alpha/K + \eta_2 m_{k}^{t-1-\tau_{f_{i,j}}} + \alpha
\]

where the superscript \( 1 : t - 1 \) means we sum over the counts along the time from \( 1 \) to \( t - 1 \). Note that we should calculate the proper distribution of Eq. (4.8) in the Block-Gibbs sampling. Finally, we compute the emission term as follows:

\[
(4.9) \quad P(a_{i,j}^t|\text{rest}) = \begin{cases} 
P(a_{i,j}^t|f_{i,j}^t = f, z_{i,j}^t = k, \phi_{i,j}^t) & \text{if adoption via social selection: } f = i, \\
P(a_{i,j}^t|f_{i,j}^t = f, z_{i,j}^t = k, \phi_{i,j}^t) & \text{if adoption via social influence: } f = i. 
\end{cases}
\]

M-step. In M-step, we minimize the negative log marginal function with respect to parameters \( \Psi, \Phi \) by using the empirical distribution of samples derived in E-step. Though we can also easily optimize other hyperparameters such as \( \alpha, \eta_i, \eta_2 \), we just evaluate them via cross validation in this paper. We formulate the optimization objective as follows:

\[
(4.10) \quad \mathcal{L} = - \log(P(a|\mathcal{H})) = -\mathbb{E}_{z,f} \left[ \sum_t \sum_i \sum_j \log P(f_{i,j}^t|\psi_i^t) \\
+ \log P(z_{i,j}^t|\text{rest}) + \log P(a_{i,j}^t|\phi_{kj}^t, \phi_{ij}^t, z_{i,j}^t = k, f_{i,j}^t = f) \\
+ \sum_t \tau_0 ||\Phi^{t+1} - \Phi^t||^2 + \sum_t \tau_0 ||\Psi^{t+1} - \Psi^t||^2 \right]
\]

For \( \Psi \), we have the following update equation:

\[
(4.10) \quad \psi_{kj}^t \propto \psi_{kj}^t - \gamma(\tau + 4\tau_0 \psi_{kj}^t - 2\tau_0 \psi_{kj}^{t-1} - 2\tau_0 \psi_{kj}^{t+1} - \frac{N^{(s)}_{ij,t}}{\psi_{ij}^t})
\]

where \( N^{(s)}_{ij,t} \) denotes the number of times \( f \) has influenced \( i \)'s decisions at time \( t \) in a single sample chain \( s \), and \( \gamma \) denotes the learning rate. After each iteration, we should normalize the vector \( \psi_i^t \).

We then consider the optimization of \( \Phi \). Note that each \( \phi_{ij}^t \) is conditioned on the vector \( \phi_k^t \). That means
each entry $\phi_{kj}^{t}$ we want to update is associated with those adoptions which are generated by $\phi_{kj}^{t}$ or $\phi_{kj}^{t-1}$. Here we let $\cdot$ in the superscript of $\phi_{kj}$ denote any user, and $\cdot$ in the subscript denote any item. We then have the following updates for $\Phi$:

$$
(\text{11}) \quad \phi_{kj}^{t} \propto \phi_{kj}^{t} - \gamma \left( - N_{kj}^{(s)} \phi_{kj}^{t} - \left\{ \frac{\partial \log \phi_{kj}}{\partial \phi_{kj}} \right\}^{(s)}_{k,j,t} \right) + \tau + 4\tau \phi_{kj}^{t} - 2\tau \phi_{kj}^{t+1} - 2\tau \phi_{kj}^{t-1}
$$

where $N_{kj}^{(s)}$ denotes the number of times an item is generated by $\phi_{kj}$ at time $t$ in a single sample chain $s$; and the symbol $\left\{ \frac{\partial \log \phi_{kj}^{t}}{\partial \phi_{kj}} \right\}^{(s)}_{k,j,t}$ denotes the sum of all the partials related to $\phi_{kj}^{t}$, and the subscript $q$ denotes any item except $j$; and

$$
\frac{\partial \log \phi_{kj}}{\partial \phi_{kj}}^{f,t} = \sum_{n=1}^{N_{kj}} \frac{\phi_{kn} \cdot \lambda(f,n,t)}{\phi_{kj} \cdot \sum_{n=1}^{N_{kj}} \phi_{kn} \cdot \lambda(f,n,t)}
$$

$$
\frac{\partial \log \phi_{kj}}{\partial \phi_{kj}}^{f,t} = - \sum_{n=1}^{N_{kj}} \phi_{kn} \cdot \lambda(f,n,t)
$$

4.1 Scalability In E-step, we linearly sweep all the adoptions in each chain. For each adoption, we jointly sample the latent influentials and interests. As a result, the complexity of each chain is $O(|\mathbf{a}| \cdot F \cdot K)$, where $F$ denotes the expected number of each user’s friends.

In M-step, we update the two matrix $\Psi^{t}$ and $\Phi^{t}$ over time. First, we can slice the time series in a much more coarse granularity for the two evolutionary parameters due to the reality that adoption data are sparse at each discrete time point. Second, though it seems that we should update all the adoptions at epoch $t$ while updating each entry in $\Phi^{t}$, in practice, we just sweep all the adoptions at epoch $t$ in a single pass, and simultaneously update the corresponding vector $\phi_{kj}^{t}$. As a result, the overall complexity of each iteration in the optimization procedure is $O(|U| \cdot F + K \cdot |V| + |\mathbf{a}| \cdot |V|)$.

Though Gibbs sampling is naturally a sequential procedure, fortunately we still can accelerate the M-step by parallelizing the updates of parameters $\psi$ and $\phi$ since the independence of each entry in $\psi$ and $\phi$. The efficiency is dependent upon the slowest machine due to the synchronous scheme.

5 Experiments
In this section we are interested in the prediction of future data and the role of the social factors played in real-world data.

5.1 Datasets We explore three kinds of publicly available real-world datasets. For diffusion purpose (preferred item factors), we consider the exact discrete ordinal time series, i.e., the timestamps in days. We hope this is without loss of generality because lots of systems do not provide the exact timestamps due to the volume of data. For the evolution purpose, we divide timestamps into epochs. The granularity of the epochs could be flexible according to the specific applications. In our following experiments, we equally divide each dataset into 10 epochs unless otherwise stated.

The first public dataset [29] is crawled from Epinions, where users can write reviews for a variety of items with ratings, and also they can add members to their trust networks or “Circle of Trust”. We treat the review behaviors as adoptions to study. Because we are mainly interested in studying the social selection and social influence phenomena, we process the raw dataset as follows. We filter those items which are totally adopted less than 10 times in the whole dataset, because less adopted items definitely have significantly little chance to be diffused along time series. We also filter those users with more than 300 friends, because it is uncommon that a user have a trust or friend list with more than 300 friends. Then we get the Epinions Dataset, which contains 14,461 users, 14,030 items, and 304,140 adoptions. Each user has a 20 friends list on average, and adopts 21 items on average. There are totally 4,307 days in this dataset.

The second public dataset [3] is crawled during Apr. 2008 to Oct. 2010 from Brightkite, where users share their check-in data with their friends. The friendship network is undirected in this dataset. We treat the check-in behaviors as adoptions to study. We use the same policy with Epinions Dataset for preprocessing Brightkite dataset. We also notice that most of the users only occur in a short period. So we only use the data recorded in the last year, and focus on analyzing the users’ behaviors in this whole year. After the filtering, we get our Brightkite Dataset, which contains 3,567 users, 10,136 items, and 888,833 adoptions. Each user has a 11 friends list on average, and adopts 249 items on average. There are totally 360 days in this dataset.

The third public dataset [14] is crawled during Nov. 2005 to Nov. 2009 from Flixster, where users can rate movies and add others to their friend lists. The friendship network is undirected. We treat whether to watch a movie as adoption behaviors to study. Same with Brightkite dataset, we only use the data from the last year. After the filtering, we get our Flixster Dataset, which contains 19,041 users, 8,251 items, and 1,393,906 adoptions. Each user has a 11 friends list on average, and adopts 74 items on average. There are totally 360
days in this dataset.

5.2 Experimental Settings

5.2.1 Experimental Data For quantitative evaluation purpose, we build the training data, golden data and test data as follows. Since our model considers the diffusion behaviors in exact discrete ordinal time series. That means current behaviors potentially could be affected by preceding behaviors. And the fewer the time intervals between two behaviors, the more opportunities they could be correlated. To alleviate such effects that could potentially bias our results, we randomly select timestamps along the time series, with the constraint that the intervals between any two selected timestamps are longer than 30. As a result, we collect all the adoptions in those selected timestamps as golden data. The rest data are served as training data. We denote golden and training data as $D_g$ and $D_{train}$, respectively. Finally we build the test data as follows:

$$D_{test} = \{a_{i,j} | a_{i,j} \in D_g \lor a_{i,j} \notin D_{train} \}$$

where $i \in u_g$ and $j \in |V|$.

where $u_g$ denotes the same set of users occurred in $D_g$.

5.2.2 Metrics Root Mean Squared Error (RMSE) is extensively used on evaluating the exact ratings users will make. However, in our problem, we consider the dyadic responses and only positive data are available, so that this is not a suitable metric in our setting. In practice, people are facing with an abundant of items, and they adopt the most interested items. Here, we consider recommending a list of $L$ items to each user and evaluate how good they are. A suitable metric in this setting is the ranking measure. For instance, the $P@N$ is used to indicate how many top-N recommended items are acquired. In our experiments, we use another popular metric, Discount Cumulative Gain (DCG), since DCG not only reflects the $P@N$ but also considers the ranking positions. The higher the acquired items are ranked, the better the method is. The DCG is defined as follows in our setting:

$$DCG = \frac{1}{TS} \sum_{t} \frac{1}{|U|} \sum_{i \in U} \sum_{l=1}^{L} \frac{I(i,l)}{\max(1,log(l))}$$

where $I(i,l)$ is the indicator function which equals to 1 if the adoption $a_{i,t}$ belongs to golden data, otherwise it equals 0. So we iterate each timestamp $t$ from a total $TS$ selected timestamps in test data, and calculate the expectation of all the users’ DCG across the timestamps. Unless otherwise stated, we set $L$ in DCG as 5 in our following experiments, because this setting is very common in modern systems.

5.2.3 Comparison Methods For comparison purpose, we implement the following methods: (1) LDA [5]. LDA is a state-of-the-art generative model for learning discrete data. It assumes that users are independent, and each user’s adoptions are dependent on the user’s own latent factors. We estimate it using an efficient Gibbs sampler [11]. (2) BPR [23]. The BPR is a state-of-the-art regression model for implicit data. BPR does not regress on the known entries of the matrix and optimize towards RMSE like many other regression models [12, 30]. BPR tries to do regression over the ranks of the items. We implement the matrix factorization of BPR and estimate it using stochastic gradient descent. BPR does not consider the social relations either. (3) SoCorr [7]. We implement the Unified Hybrid method which is one of the most competitive models discussed in [7]. It employs the social correlation for the recommendation. It assumes each user $i$’s adoption is dependent on latent factors of user $i$, $i$’s influential, and item $j$. It’s also an generative model. We do not report other social recommendation methods because they are not learned for dyadic responses, and they assume the influential relations are already known.

5.2.4 Parameter Settings The hyperparameters $H = \{\alpha, \eta\}$ are set by validation on different datasets. In the following experiments, we set them as $\alpha = 1.0, \eta_1 = 0.05, \eta_2 = 0.1$. We generate 20 samples in each E-step and iterate 20 times in M-step.

5.3 Evaluations

5.3.1 Number of Factors We measure the DCG results of LDA, BPR, SoCorr, and LFDM, while varying the number of latent factors (interests) $K$.

We show the results in Figure 1. It can be seen that our model outperforms all the other methods on all the factors in the three datasets. Note that the relative results on Epinions Dataset is much lower than results on Brightkite Dataset and Flixster Dataset because the number of each user’s adoptions at each timestamp in Epinions Dataset is much sparser than the other two datasets. Though SoCorr models the latent influence spaces, it actually does not distinguish the cascading influence diffusion and users’ own tastes as our model. They just smooth over all friends’ latent user factors by weighting the influence strength. As a result, our model LFDM performs better than SoCorr especially on Brightkite Dataset and Flixster Dataset where social influence plays an relatively important role.
in each epoch. It can be seen that 10 — 20 epochs would span too few timestamps since the data sparsity is built in. We notice that users from Epinions can tend to infect each other even though a trust network is not drawn on the convergence rate of LDA or BPR because they are estimated using either Gibbs sampling or stochastic gradient descent, thus they are actually not comparable with the EM-styled algorithms developed in LFDM or SoCorr. The results show that our algorithm converges fast on each dataset within 50 iterations.

5.3.4 Convergence Rate We show the held-out likelihoods of all the methods in Figure 3. Our model gives higher likelihoods than others. In addition, we also depict the convergence rate of LFDM and SoCorr. We do not draw the convergence rate of LDA or BPR because they are estimated using either Gibbs sampling or stochastic gradient descent, thus they are actually not comparable with the EM-styled algorithms developed in LFDM or SoCorr. The results show that our algorithm converges fast on each dataset within 50 iterations.

5.3.2 Influential Analysis We investigate the prediction accuracy for groups of users with different influential relations. The diagonal values $\psi_{i,i}$’s indicate the tendency that user $i$ prefers to adopt an item that meets his or her own tastes. That means the possibility of user $i$ to be influenced is $\psi_{i,i}$. We bin users from test data into three groups with low as $\psi_{i,i} \in [0, 1/10)$, mid as $\psi_{i,i} \in (1/10, 4/10)$ and high as $\psi_{i,i} \in (4/10, 1)$. We show the DCG results of the three groups in Figure 2. It can be seen that our method significantly makes better predictions for users who are more likely to be influenced in Brightkite and Flixster datasets.

5.3.3 Evolution Analysis We study the granularity of epochs and its effects on our results. We equally divide each dataset into 1, 10, 20, 50 epochs respectively. We evaluate the average DCG measures on each of the datasets as before, and show the results in Table 2. It’s hard to characterize the evolving parameters if each epoch spans too few timestamps since the data sparsity in each epoch. It can be seen that 10 — 20 epochs would be good choices for our datasets.

5.3.5 Tendency to be Influenced We use LFDM to illustrate the user proportions of each online services according to users’ tendency to be influenced in Figure 4. Surprisingly, most of users in Epinions do not tend to infect each other even though a trust network is built in. We notice that users from Epinions can
write reviews that may earn them money and recognition under a well-established reputation system. This may be a major motivation for users to establish their friendship networks. In contrast, users from Brightkite are relatively more likely to be influenced. This may because the location information is more likely to be shared among intimates, and intimates potentially have stronger influence on each other. We also investigate users’ influence relations $\Psi$ estimated by SoCorr. We find that SoCorr tends to estimate more than 60% of adoptions due to influence. However, in reality there is no such pervasive cascading patterns in those datasets. So it potentially bias the results.

6 Conclusion

In this paper we discussed the modeling of users’ adoption behaviors by considering both social selection and influence. We presented a novel probabilistic LFDM model which incorporates a generalized diffusion model into a latent factor framework. We further developed an effective EM-styled algorithm on estimating LFDM. We evaluated the model goodness using a practical metric (DCG) on three kinds of user adoption data. We also analyzed the characteristics of those real-world datasets using our method. As the explosive growth of social media, such analysis can be used to improve lots of personalized services. In future work we would further analyze both social selection and influence toward point processes so that we could extend the study of adoption behaviors to continuous time series.

References