Collaborative Dynamic Sparse Topic Regression with User Profile Evolution for Item Recommendation

Li Gao,[*] Jia Wu,[†] Chuan Zhou,[*‡] Yue Hu[*‡]
[*] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[†] Quantum Computation & Intelligent Systems Centre, University of Technology Sydney, Australia
[‡] University of Chinese Academy of Sciences, Beijing, China
{gaoli, zhouchuan, huyue}@iie.ac.cn, jia.wu@uts.edu.au

Abstract

In many time-aware item recommender systems, modeling the accurate evolution of both user profiles and the contents of items over time is essential. However, most existing methods focus on learning users' dynamic interests, where the contents of items are assumed to be stable over time. They thus fail to capture the dynamic changes in the item's contents. In this paper, we present a novel method CDUE for time-aware item recommendation, which captures the evolution of both user's interests and item's contents information via topic dynamics. Specifically, we propose a dynamic sparse topic model to track the evolution of topics for changes in items' contents over time and adapt a vector autoregressive model to profile users' dynamic interests. The item's topics and user's interests and their evolutions are learned collaboratively and simultaneously into a unified learning framework. Experimental results on two real-world data sets demonstrate the quality and effectiveness of the proposed method and show that our method can be used to make better future recommendations.

Introduction

Many recommendation applications, such as Netflix's movie recommendation and scientific article recommendation, provide item recommendation based on a user's interests and the contents of item (such as its descriptions and attributes). Currently, methods for learning users' interests through combining contents information have proven to be promising for tackling this issue (Purushotham, Liu, and Kuo 2012; Wang, Chen, and Li 2013; Chen et al. 2014; Bansal, Das, and Bhattacharyya 2015; Lu et al. 2015; Wang, Wang, and Yeung 2015). For example, collaborative topic regression (Wang and Blei 2011) combines traditional collaborative filtering with topic modeling, which represents users with topic interests and assumes item contents are generated by a topic model. However, all these models assume that users have constant interests toward the item set and item contents do not change once formulated, which is only true in the off-line settings.

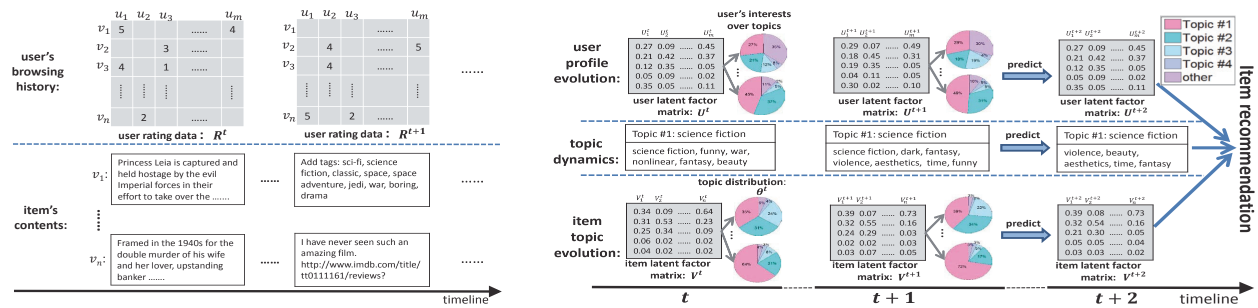
In deed, in many real-world scenarios, a user's interests toward some items may drift over time since they may be affected by moods, contexts, and pop culture trends (Li et al. 2011; Aly et al. 2013; Liu 2015; Du et al. 2015). For example, a user falling in love may start to purchase romantic gifts, while she has never behaved like that before. Moreover, the contents of items in online platforms, such as blogs and social networking sites, may also change over time (Zhang, Kim, and Xing 2015; Wang and Klabjan 2016; Dai et al. 2016). For example, in product recommender systems with a wealth of online consumer-generated data, different users may add comments or product reviews over time. As another example, bloggers in the blogosphere may pursue hot topics to meet current users' interests to increase their followings.

Thus, to provide timely item recommendation, it is essential to capture users' dynamic interests and temporal changes in contents of items. A straightforward way to model a user's dynamic interests is to build an individual matrix factorization model (Gao et al. 2016) to profile the user's interests learned from the data set at each time interval, then discover its evolutionary patterns using time series analysis or sequential pattern mining (Lu et al. 2016). To analyze text data stream, dynamic topic models have been proposed by a large number of works (Blei and Lafferty 2006; Kurashima et al. 2013; Zhang, Kim, and Xing 2015; Gad et al. 2015). However, to the best of our knowledge, no previous work has jointly considered both temporal evolutionary factors via topic dynamics for time-aware item recommendation.

In this paper, we propose a novel method CDUE for item recommendation that combines the dual evolutions of both item topics and user profiles. Specifically, we introduce a vector autoregressive model to capture the users' temporal profiles that reflect their interests, and we propose a dynamic sparse topic model to track the dynamic topics in an item's contents that change over time. We jointly learn users' interests and sparse topic modeling of items in a collaborative filtering fashion. Then we model the evolutions of users' profiles and sparse topics of the items to further guide iterative learning of both in the future. Based on the predicted user's interests and item's sparse topics, we can provide recommendations at a certain future time interval.

The main contributions of this paper are summarized as follows.

- We propose a new item recommendation problem by jointly considering the evolution of both item topics and user profiles over time.



(a) Input: users' browsing history and the contents of items over time.

(b) Output: temporal evolution of item topics and user profiles.

Figure 1: Problem statement. (a) Input is the browsing history of users and the contents of items over time. (b) As output we aim at capturing the evolution of both item topics and user profiles via topic dynamics over time. We then provide recommendations at a certain future time interval.

- We develop a principled method CDUE to integrate vector autoregressive and dynamic topic modeling of contents information and matrix factorization into a unified learning framework, where users' interests and items' topics and their evolutions are learned collaboratively and simultaneously.
- We develop a novel dynamic topic model to track the evolution of sparse topics in items, where we claim topic sparsity and use topic regression to distinguish topics that explain recommendations from topics that are important for explaining contents.
- We compare CDUE with the state-of-the-art methods on two real-world data sets to demonstrate the performance and effectiveness. Our method also shows good interpretive ability with topic dynamics for time-aware item recommendation.

Problem Statement

We denote the user set by $\mathcal{U} = \{u_1, u_2, \ldots, u_m\}$, and the item set by $\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$. For a time interval t, the users' browsing history is recorded by an $m \times n$ matrix R^t. R^t_{ij} is the value of the cell (i, j) of R^t, which is either a rating given by u_i on v_j or a missing value ($R^t_{ij} = 0$ in this case). We assume each item v is associated with a "bag-of-words", which is constructed from the relevant textual contents such as its description, and assume the item contents are generated by a topic model. The summation of these texts constructs a corpus, which is used for topic modeling.

Given a sequence of users' browsing behavior with T time intervals, $R^1, R^2, \ldots, R^T$, we aim to combine the topic distribution θ^t of items with the users' K-dimensional interests $U^t \in \mathbb{R}^{K\times m}$ in each of the time intervals to learn the users' browsing behavior. The θ^t tracks the temporal evolution of the topics for the item v, and U^t explores the user profiles' evolutionary processes. We can then make recommendations to users at future time intervals $\{T+1, T+2, \ldots\}$ based on the predicted topic distribution of items and users' interests.

Modeling Dual Evolutions

In this section, we model the evolution of both user profiles and item topics over time.

Modeling User Profile Evolution

Modeling the evolution of users' profiles over time is essential for recommender systems (Liu 2015; Dai et al. 2016). A proper item recommendation should not only consider the users' current interests, but also the way their interests might evolve in the future. We model the changes in user's interests adapting a vector autoregressive model (Zivot and Wang 2006), which gives promising performance in recommendation (Lu et al. 2016).

Based on a vector autoregressive model, the user's latent factor vector $U_i^t \in \mathbb{R}^K$ in the time interval t can be modeled by the factor vectors of the user in the previous τ ($\tau < t$) time intervals:

$$U_i^t = C_i^1 U_i^{t-1} + C_i^2 U_i^{t-2} + \cdots + C_i^\tau U_i^{t-\tau} + \epsilon_i^t \quad (1)$$

where $\{C_i^j \in \mathbb{R}^{K\times K}\}_{j=1}^{\tau}$ are the coefficient matrices, and $\epsilon_i^t \in \mathbb{R}^K$ is uncorrelated Gaussian noise with a zero-mean and a time invariant covariance matrix $D_i \in \mathbb{R}^{K\times K}$.

With a given τ and an available series of user interests of T time intervals $\{U_i^t\}_{t=1}^T$ for user i, we use a least square estimation to learn the parameters $\{C_i^j\}_{j=1}^{\tau}$ and D_i in Eq. (1). The details of the mathematical derivation follow the work (Neumaier and Schneider 2001). We can then predict the user's future interests $\{U_i^t\}_{t=T+1}$ based on the observed time series of the user's interests $\{U_i^t\}_{t=1}^T$ and the learned parameters.

Modeling Item Topic Evolution

We model the evolution of item topics adapting the sparse topical coding framework (Zhu and Xing 2011), which is a topic model that directly controls the posterior sparsity. It makes intuitive sense that each item's contents and word would only be associated with a few salient topic meanings. The work (Zhang, Kim, and Xing 2015) has shown that sparse topical coding can properly capture the robust text representation of the contents in the online platforms.

We use the *tf-idf* weighted bag-of-words model for the items' contents descriptor, where we build a dictionary of item vocabularies after filtering the stop words and removing words occurred fewer than 50 times. Each item's contents can be represented as a vector. Let $w_j = \{w_{j1}, w_{j2}, \ldots, w_{j|N_j|}\}$ be the item feature vector for v_j, where $|N_j|$ is the index set of words and w_{jd} ($d \in N_j$) represents the number of appearances of word d in item v_j.

Let $\beta \in \mathbb{R}^{K \times N}$ be the matrix of K-topic bases for each item word, where N is the size of the vocabulary. We assume that each row $\beta_{k.}$ indicates the k-th topic distribution over the vocabulary. We denote $\theta_j \in \mathbb{R}^K$ as the document code of item v_j, which is the latent topic distribution of v_j. Let $z_{jd} \in \mathbb{R}^K$ be the word code that denotes the latent topic representation of individual item word d in v_j.

Then, for the contents of item v_j, the generative process is described below,

1. Sample a latent topic distribution θ_j from a prior $p(\theta)$.
2. For each observed word d of item v_j,
   (a) Sample a word code $z_{jd} \sim p(z_{jd}|\theta_j)$.
   (b) Sample an observed item word feature $w_{jd} \sim p(w_{jd}|z_{jd}, \beta)$.

The distributions used in the above process are defined as follows: to achieve sparsity on θ and z, we choose the prior $p(\theta) \propto \exp(-\lambda\|\theta\|_1)$, and we define $p(z_{jd}|\theta_j)$ as a composite distribution $p(z_{jd}|\theta_j) \propto \exp(-\delta\|z_{jd} - \theta_j\|_2^2 - \rho\|z_{jd}\|_1)$. To generate item word features, we choose to apply a Gaussian distribution with a mean of $z_{jd}^{\mathcal{T}}\beta_{.d}$ to the item word feature to make optimization easier and the model applicable to rich forms of data, *i.e.*, $p(w_{jd}|z_{jd}, \beta) = \mathcal{N}(w_{jd}; z_{jd}^{\mathcal{T}}\beta_{.d}, \sigma_w^{-1})$, where $\beta_{.d}$ is the d-th column of β.

Temporal Evolution of Topics In order to model the temporal evolution of the topics for each item, using a dynamic topic model (Blei and Lafferty 2006; Zhang, Kim, and Xing 2015), we let β change over time. By following the state space model with Gaussian noise, we evolve β^t from β^{t-1}. Thus, for each topic k, we have

$$p(\beta_{k.}^t|\beta_{k.}^{t-1}) = \mathcal{N}(\beta_{k.}^{t-1}, \sigma_\beta^{-1}I) \tag{2}$$

Collaborative Dynamic Sparse Topic Regression with User Profile Evolution

In this section, we describe our proposed method CDUE for item recommendation.

Collaborative Dual Evolutions

We assume that a user's factor vector U_i represents user u_i with topic interests. CDUE fits a model that uses the latent topic space and the users' interests to explain both the observed ratings and the observed item words. In addition, we introduce a latent variable ϵ_j to offset the topic proportions θ_j when modeling the ratings. Thus, for the time interval t, the rating R_{ij}^t can be formulated as

$$R_{ij}^t \sim \mathcal{N}(U_i^{t\mathcal{T}}(\theta_j^t + \epsilon_j^t), {\sigma_{ij}^t}^{-1}) \tag{3}$$

where the item latent vector V_j^t is set as $V_j^t = \theta_j^t + \epsilon_j^t$. The item latent offset ϵ_j, $\epsilon_j \sim \mathcal{N}(0, \sigma_v^{-1}I)$, can force our model to distinguish topics that explain recommendations from topics that are important for explaining item contents. The variable $\sigma_{ij}^t = a$, if u_i has rated v_j in t; otherwise, $\sigma_{ij}^t = b$, where $a > b > 0$. A similar setup is used in the work (Wang and Blei 2011).

Note that before applying the vector autoregressive method to model the evolutionary process of users' interests, as shown in Eq. (1), we should first obtain the initial sequence of user's latent factor vectors to meet the conditions. Suppose T_0 is the number of the initial user's latent factor vectors. For the time interval t, we choose to learn the latent matrix U^t as follows:

$$p(U^t|A^t, \Omega) = \prod_{i=1}^{m} \mathcal{N}(A_i^t, \sigma_u^{-1}I) \tag{4}$$

where A_i^t is calculated as:

$$A_i^t = \begin{cases} 0, & \text{if } t \leq T_0 \\ C_i^1 U_i^{t-1} + \cdots + C_i^\tau U_i^{t-\tau}, & \text{if } t > T_0 \end{cases} \tag{5}$$

Based on the above, given the sequence of user browsing behavior $R^t (t = 1, \ldots, T)$ and the hyperparameter set $\{\lambda, \delta, \rho, \sigma_w, \sigma_\beta, \sigma_v, \sigma_{ij}, \sigma_u\}$, maximizing the posterior is equivalent to minimizing the complete negative log likelihood of $\Theta = \{\theta_j, z_j\}_{j=1}^n$, β, V, U. We accumulate the negative log likelihoods of all time ranges, and seek an optimal solution for all time intervals that yields a more accurate fitness of the data. So the objective of CDUE is derived as

$$\begin{aligned} \min_{\{\Theta^t, \beta^t, V^t, U^t\}_{t=1}^T} & \sum_{t=1}^T \sum_{i=1}^m \sum_{j=1}^n \sigma_{ij}^t (R_{ij}^t - U_i^{t\mathcal{T}} V_j^t)^2 \\ & + \sum_{t=2}^T \sigma_\beta \|\beta^t - \beta^{t-1}\|_2^2 + \sum_{t=1}^T \sum_{i=1}^m \sigma_u \|U_i^t - A_i^t\|_2^2 \\ & + \sum_{t=1}^T \sum_{j=1}^n (\lambda\|\theta_j^t\|_1 + \sigma_v \|V_j^t - \theta_j^t\|_2^2) \\ & + \sum_{t=1}^T \sum_{j=1}^n \sum_{d \in N_j^t} (\delta\|z_{jd}^t - \theta_j^t\|_2^2 \\ & \qquad + \rho\|z_{jd}^t\|_1 + \sigma_w (w_{jd}^t - {z_{jd}^t}^{\mathcal{T}} \beta_{.d}^t)^2) \\ \text{s.t.} \quad & \theta_j^t \geq 0, \forall j, t. \;\; z_{jd}^t \geq 0, \forall j, d, t. \;\; \beta_{k.}^t \in \mathcal{P}, \forall k, t \end{aligned} \tag{6}$$

where N_j^t denotes the word index set of item v_j in time interval t, and $\mathcal{P}$ is the $(N-1)$-simplex (*i.e.*, we set the sum of $\beta_{k.}^t$ as one for $\forall k, t$).

Discussion As shown in Eq. (6), we choose to globally learn the objective over the data in all time intervals, which is less scalable but generates more accurate fitness of the data. As an alternative, we can seek a local minimum in the current time interval t and learn the objective from the data up to $t-1$, which is more practical in a real-world scenario, and can be easily extended from Eq. (6). Due to space limitations, we skip the derivation.

Proposition 1. *The objective of Eq.* (6) *is multi-convex, i.e., the objective is convex over one parameter set when the others are fixed. Moreover, the feasible set is a convex set.*

Parameter optimization

Given Proposition 1, a natural way to solve the optimization problem in Eq. (6) is coordinate descent. Note that in Eq. (6), every two adjacent time intervals are only coupled by the parameter β. If we fix β, the objective for each time interval becomes independent of one another. Thus, we alternate the parameter optimization between β and the parameters of the other variables. Specifically, the procedure alternatively performs as outlined below.

1. Fix $\{\beta^t\}_{t=1}^T$ to optimize over $\{\Theta^t, V^t, U^t\}_{t=1}^T$. We first decouple the optimization of every time interval t. Then, we can further employ coordinate descent to alternately optimize $\Theta^t = \{\theta_j^t, z_j^t\}_{j=1}^n$ and V^t, U^t.

(a) While fixing Θ^t, *i.e.*, given the current estimate of Θ^t, we aim to optimize U_i^t and V_j^t to solve the objective

$$\min_{\{V^t,U^t\}} \sum_{i=1}^m \sum_{j=1}^n \sigma_{ij}^t (R_{ij}^t - {U_i^t}^{\mathcal{T}} V_j^t)^2 + \sum_{i=1}^m \sigma_u \|U_i^t - A_i^t\|_2^2 + \sum_{j=1}^n \sigma_v \|V_j^t - \theta_j^t\|_2^2 \tag{7}$$

Note that the value of A_i^t has been updated by the Eq. (5). Taking the gradient with respect to U_i^t and V_j^t, and setting it to zero leads to

$$\begin{aligned} U_i^t &\leftarrow (\sum_{j=1}^n \sigma_{ij}^t V_j^t {V_j^t}^{\mathcal{T}} + \sigma_u I)^{-1} (\sum_{j=1}^n \sigma_{ij}^t V_j^t R_{ij}^t + \sigma_u A_i^t) \\ V_j^t &\leftarrow (\sum_{i=1}^m \sigma_{ij}^t U_i^t {U_i^t}^{\mathcal{T}} + \sigma_v I)^{-1} (\sum_{i=1}^m \sigma_{ij}^t U_i^t R_{ij}^t + \sigma_v \theta_j^t) \end{aligned} \tag{8}$$

(b) While fixing V^t and U^t, we aim to optimize the parameters Θ^t, *i.e.*, $\{\theta_j^t, z_j^t\}_{j=1}^n$. Since the items can be assumed to be independent of one another, we can perform this step for each item separately by solving

$$\begin{aligned} &\min_{\theta_j^t, z_j^t} (\lambda \|\theta_j^t\|_1 + \sigma_v \|V_j^t - \theta_j^t\|_2^2) \\ &+ \sum_{d \in N_j^t} (\delta \|z_{jd}^t - \theta_j^t\|_2^2 + \rho \|z_{jd}^t\|_1 + \sigma_w (w_{jd}^t - {z_{jd}^t}^{\mathcal{T}} \beta_{.d}^t)^2) \\ &\text{s.t.} \quad \theta_j^t \geq 0. \ \ z_{jd}^t \geq 0, \forall d. \end{aligned} \tag{9}$$

We can then solve the convex problem with coordinate descent by alternately optimizing θ_j^t and z_j^t, all of which have closed-form solutions.

- Optimize over z_j^t when θ_j^t is fixed. Let $f(\theta_j^t, z_j^t)$ denote the objective of Eq. (9), where z_{jd}^t ($d \in N_j^t$) is not coupled while fixing θ_j^t. Given Proposition 2, the solution is $z_{jd}^t = \max(0, \tilde{z}_{jd}^t)$, where each dimension of $\tilde{z}_{jd}^t$ is compared to 0, $\tilde{z}_{jd}^t = \text{argmin}_{z_{jd}^t} f(\theta_j^t, z_j^t)$. By setting the gradient $\nabla_{z_{jd}^t} f(\theta_j^t, z_j^t) = 0$, we have

$$\tilde{z}_{jd}^t = (\delta I + \sigma_w \beta_{.d}^t {\beta_{.d}^t}^{\mathcal{T}})^{-1} (\delta \theta_j^t - \frac{\rho}{2} I + \sigma_w \beta_{.d}^t w_{jd}^t) \tag{10}$$

- Optimize over θ_j^t when z_j^t is fixed. Given Proposition 2, the solution is $\theta_j^t = \max(0, \tilde{\theta}_j^t)$, where $\tilde{\theta}_j^t = \text{argmin}_{\theta_j^t} f(\theta_j^t, z_j^t)$ and each dimension of $\tilde{\theta}_j^t$ is compared to 0. Setting the gradient $\nabla_{\theta_j^t} f(\theta_j^t, z_j^t) = 0$, we have the solution

$$\theta_j^t = \max\{0, \frac{\delta \sum_{d \in N_j^t} z_{jd}^t + \sigma_v V_j^t - \frac{\lambda}{2} I}{\sigma_v + \delta |N_j^t|}\} \tag{11}$$

where $|N_j^t|$ is the number of words in item v_j in time interval t. Note that when $\sigma_v = 0$, the optimal $\hat{\theta}_j^t$ is the truncated average of z_{jd}^t.

2. Fix all the parameters $\{\Theta^t, V^t, U^t\}_{t=1}^T$ to optimize over $\{\beta^t\}_{t=1}^T$. This step aims to optimize

$$\begin{aligned} &\min_{\{\beta^t\}_{t=1}^T} \sum_{t=2}^T \sigma_\beta \|\beta^t - \beta^{t-1}\|_2^2 \\ &+ \sum_{t=1}^T \sum_{j=1}^n \sum_{d \in N_j^t} \sigma_w (w_{jd}^t - {z_{jd}^t}^{\mathcal{T}} \beta_{.d}^t)^2 \\ &\text{s.t.} \quad \beta_{k.}^t \in \mathcal{P}, \forall k, t \end{aligned} \tag{12}$$

We can obtain the solution using coordinate descent and projected gradient descent, where we solve every β^t one by one for each time interval t, *i.e.*, we fix $\{\beta^t\}_{t=1}^T \backslash \beta^t$, We then employ a Euclidean projection method onto the positive simplex (Duchi et al. 2008) to solve β^t at every iteration. Specifically, we have

$$\beta_{kd}^t = \max\{\tilde{\beta}_{kd}^t - \eta, 0\} \tag{13}$$

$$\tilde{\beta}_{kd}^t = \frac{\sigma_\beta \beta_{kd}^{t-1} + \sigma_w \sum_{j=1}^n z_{jdk}^t (w_{jd}^t - \sum_{c \neq k} z_{jdc}^t \beta_{cd}^t)}{\sigma_w \sum_{j=1}^n (z_{jdk}^t)^2 + \sigma_\beta} \tag{14}$$

where $\eta = \frac{1}{\xi}(\sum_{i=1}^\xi \varphi_i - 1)$. The vector φ is obtained by sorting $\tilde{\beta}_{k.}^t$ in a descending order. The value of ξ is $\xi = \max\{l \in [N] : \varphi_l - \frac{1}{l}(\sum_{r=1}^l \varphi_r - 1) > 0\}$, where $[N]$ is the index set. Note that the value of β_{kd}^1 is obtained by ignoring the terms σ_β and $\sigma_\beta \beta_{kd}^{t-1}$. The projection onto the simplex $\mathcal{P}$ can be performed with a linear algorithm (Duchi et al. 2008).

Proposition 2. *Let $f(x)$ be a strictly convex function. The optimum solution x^* of the constrained problem: $min_{x \geq 0} f(x)$ is $x^* = max\{0, x_0\}$, where x_0 is the solution of the unconstrained problem $min_x f(x)$ (Zhu and Xing 2011).*

Prediction

Given a sequence of users' browsing behavior R^t and items with relevant textual contents for the time interval t ($t = 1, \ldots, T$), based on CDUE, we can learn: the user profile evolution coefficient matrices $\{C^l\}_{l=1}^{\tau}$; the user latent interests $\{U^t\}_{t=1}^T$; the item latent matrices $\{V^t\}_{t=1}^T$; the K-topic bases $\{\beta^t\}_{t=1}^T$; and the item latent topic distributions $\{\theta^t\}_{t=1}^T$.

We can then predict item recommendation for users at a particular future time interval $T + \iota$ (usually ι is set as 1). We compute $A^{T+\iota}$ by recursively applying Eq. (4) and Eq. (5). Based on Eq. (2) and the items' textual contents, we sample β^t step by step to obtain $\beta^{T+\iota}$ and $\theta^{T+\iota}$, where $\mathbb{E}[\epsilon^t] = 0$ and $V^t = \theta^t$, since no user browsing behavior R^t are obtained in the process ($t = \{T+1, \dots, T+\iota\}$). We then use the point estimate of $U_i^{T+\iota}$ and $\theta_i^{T+\iota}$ to make prediction on the rating of user u_i on item v_j: $\tilde{R}_{ij}^{T+\iota} \approx {U_i^{T+\iota}}^{\mathcal{T}} \theta_j^{T+\iota}$. Thus, recommendations on which items the user u_i would be interested in are made by ranking the predicted ratings.

Experiments

Data Sets

We select two data sets extracted from the work (Cantador, Brusilovsky, and Kuflik 2011) for analysis.

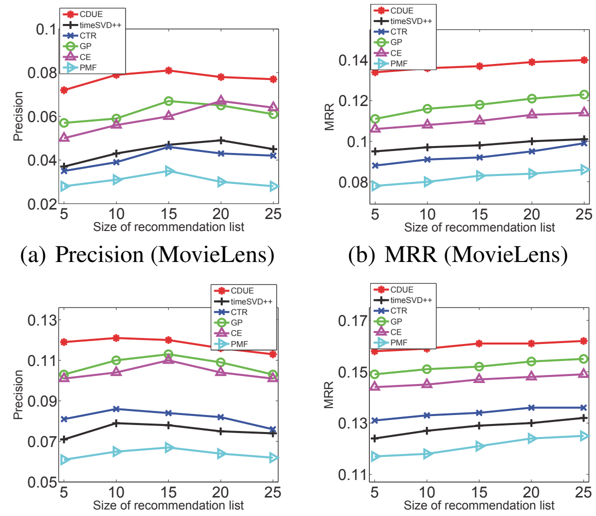
The first data set was collected from MovieLens (https://movielens.org/). It contains $2,113$ users, $10,197$ movies, $13,222$ tags, $855,598$ movie rating records and $47,957$ tag assignments provided by users with timestamps. The second data set was collected from Last.fm (http://www.last.fm), which records $1,892$ users, $17,632$ music artists, $92,834$ user-listened artist relations, $11,946$ unique tags and $186,479$ tag assignments with timestamps. Similar to the work (Liu 2015), for each data set, we treat each item, *i.e.*, movie and music artist, as a document and the associated tags with timestamps as terms for dynamic topic modeling.

Experimental Settings

Comparison Methods We compare our proposed method CDUE with the following methods:

- PMF (Salakhutdinov and Mnih 2007). Probabilistic matrix factorization (PMF) is an effective MF method for item recommendation.
- timeSVD++ (Koren 2009). timeSVD++ tracks the time changing behavior throughout the life span of the data, which extends the SVD++ factor model by incorporating time changing parameters.
- CTR (Wang and Blei 2011). Collaborative topic regression (CTR) is a standard content-based recommendation method that combines latent factor models with content analysis based on probabilistic topic modeling.
- GP (Liu 2015). GP models users' dynamic preferences for personalized recommendations, where a Gaussian process is applied to predict the user's preference in the next interaction.
- CE (Lu et al. 2016). Collaborative evolution (CE) proposes learning the evolution of user profiles through the historical data of recommendations and outputs a prospective user profile for the future.

Metrics We adopt the following two metrics to measure the recommendation performance, namely Precision@n (P@n) and Mean Reciprocal Rank (MRR), where P@n measures the ratio of successfully recommended items to the top-n recommendation and MRR measures the reciprocal of the first occurrence position of the ground truth item for each user (Liu 2015; Zhang and Wang 2015). The two metrics are first calculated separately on each user's recommendation list and then taken an average among all test users. The higher values of the two metrics are favored in comparisons.



(a) Precision (MovieLens) (b) MRR (MovieLens)

(c) Precision (Last.fm) (d) MRR (Last.fm)

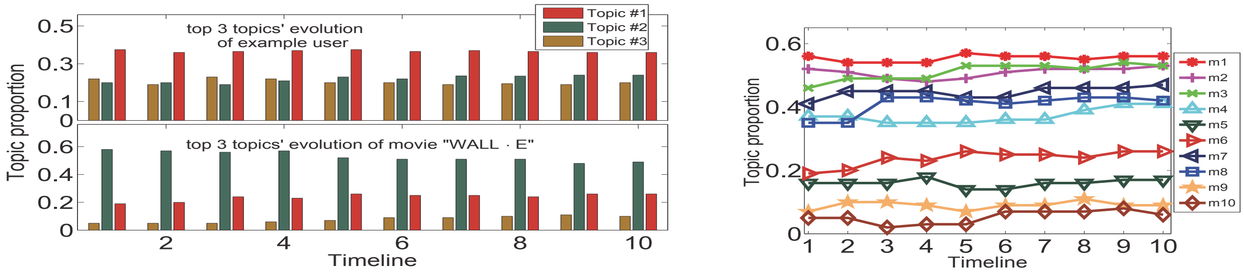
Figure 2: Experimental results of the compared methods on the two data sets.

Settings For both data sets, we split the data over time. Data within a continuous T time intervals are used for training and subsequent data are for testing. We then randomly choose one record from the most recent time interval from the training data set to construct the validation data. The P@10 performance on the validation data for each data set is used to select the optimal parameters. We use 10 topics for all topic modeling based models and the value of τ is set as $\tau = 5$. As a result, we set the hyperparameters as $a = 1$, $b = 0.01$, $\lambda = 0.1$, $\sigma_v = 1$, $\rho = 10$, $\sigma_w = 1$, $\sigma_\beta = 1$, $\sigma_u = 1$, $\delta = 0.1$. For each model, we run the experiments 10 times and show the averaged results. We also conduct paired t-tests (p-value < 0.05) to confirm that all results are statistically significant.

Experimental Results

Figure 2 shows the experimental precision and MRR results of the comparisons with respect to a range of recommendation list sizes ranging from 5 to 25 with 5 as the increment, where the size of time intervals for training is set as $T = 30$. From the figure, we see that timeSVD++ outperforms the basic PMF method for both data sets, demonstrating the importance of modeling the temporal dynamics. Even though no temporal information is considered, CTR outperforms timeSVD++ on the Last.fm data set, while timeSVD++ performs better on the MovieLens data set. GP and CE consistently outperform timeSVD++ and CTR, which proves that modeling the evolution of users' preferences can further improve time-aware recommendation performance.

In all cases, CDUE significantly outperforms the baselines. Compared with GP, we jointly learn the users' pref-



(a) Top 3 topics' evolution of the example user and the movie "WALL · E" (the first correctly predicted movie) along the timeline.

(b) Variation of topic proportion of the top 10 predicted movies over Topic #1 (*i.e.*, the top 1 topic).

Figure 3: The evolution of the top 3 topics for the example user and the top 10 predicted movies (as shown in Table 2) from September 23, 2008 to December 31, 2008 with 10 days as the duration of each time interval. Note that the value of the topic proportion in the figure is normalized such that the sum of entries of the vector U_i^t and θ_j^t, if is not 0, is equal to 1. The values of topic proportion for the last time interval are predicted by our model based on historical data.

Table 1: Average performance of the compared methods over a range of recommendation list sizes.

	MovieLens		Last.fm	
	Precision	MRR	Precision	MRR
PMF	0.0304	0.0822	0.0638	0.1210
CTR	0.0410	0.0928	0.0818	0.1340
GP	0.0618	0.1178	0.1076	0.1522
CE	0.0594	0.1102	0.1040	0.1466
timeSVD++	0.0442	0.0982	0.0754	0.1284
CDUE	**0.0774**	**0.1372**	**0.1178**	**0.1602**

erence evolution and topic modeling of items in a collaborative filtering fashion. With respect to the CE that enforces the factor matrices for items to be shared over different time intervals, we further model item topic evolution using dynamic topic modeling. It is worth noting that CDUE can be adapted to formulate CE model, *i.e.*, CDUE is a general model that can capture more temporal effects than CE for timely item recommendation. Moreover, Table 1 summarizes the P@n and MRR performance of the compared methods, averaged over different recommendation list sizes, which shows a similar result to the above.

Discussion Aside from promising prediction performance, our recommendation model also provides good interpretive ability using the topics learned from the data. Table 2 shows an example user in a particular time interval (ranging from December 22, 2008 to December 31, 2008, where the duration of each time interval is 10 days) with top 10 preferred movie recommendations as predicted by CDUE and the top 3 matched topics that are found by ranking the entries of her predicted latent vector U_i^t. We use several top representative words to represent the topic. The learned topics serve as a summary of her interests. From Table 2, we see that the user might have more interest in the movies that are romantic, fantasy and comic. The value of P@10 is 0.2.

We further show the user's interests using the entries of her latent vector U_i^t for each time interval, the item's topical representation using the latent topic distribution θ_j^t of the item, and track their evolutions over the topics along

Table 2: One example user (userID: 6757) in a particular time interval (ranging from December 22, 2008 to December 31, 2008), with the top 3 matched topics and predicted top 10 preferred movies. The last column shows whether the user has seen the movie.

top 3 topics	1. romance, memory, classic, fashion, love, nudity(topless), oscar(best picture), sex	
	2. surreal, fantasy, time travel, space, science fiction, surrealism, intelligent	
	3. comedy, classic, funny, twist ending, drama, robin williams, romance, action	
top 10 movies	1. Ever After	no
	2. City Lights	no
	3. Benny & Joon	no
	4. The Science of Sleep	no
	5. Willy Wonka & the Chocolate Factory	no
	6. WALL · E	yes
	7. American Pie	no
	8. She's All That	no
	9. Back to the Future Part III	no
	10. Burn After Reading	yes

the timeline. Due to space limitations, we only report the evolving chain of the top 3 matched topics for the example user and the corresponding top 10 predicted movies as shown in Table 2 from September 23, 2008 to December 31, 2008 with 10 days as the increment.

Figure 3 shows the experimental results. Our method successfully captures the evolution of the user's interests and the item's contents information via topic dynamics. Figure 3(a) illustrates that the example user consistently has the highest interest in Topic #1 and shows more interest in Topic #2 than Topic #3 along with time. The topic proportion of Topic #1 for the movie "WALL · E" gradually increases along the timeline, possibly due to the fact that more tags about romance, such as *love story*, are added to this movie. In the last time interval (ranging from December 22, 2008 to December 31, 2008), the movie "WALL · E" shows higher topic proportions of Topic #1 and Topic #2, thus can meet the example user's taste with higher probability. As shown in Figure

3(b), as more tags are added to the movies, the movies display dynamic ups-and-downs over Topic #1, which demonstrates that the item has dynamic topic distribution over time and it is necessary to consider the topic evolution of contents information for time-aware item recommendation.

Conclusions

In this paper, we study a new item recommendation problem that captures the evolution of both user's interests and item's contents via topic dynamics. A novel model CDUE is proposed that integrates user profile evolution and dynamic topic modeling of contents information and matrix factorization into a unified learning framework. Compared to the baselines, our method significantly obtains the better performance in terms of the Precision and MRR metrics. Thus, our model can be used to make better future recommendations.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable comments and suggestions. This work was supported by the 973 project (No. 2013CB329605), NSFC (No. 61502479 and 61602466) and Strategic Leading Science and Technology Projects of CAS (No. XDA06030200). C. Zhou is the corresponding author.

References

Aly, M.; Pandey, S.; Josifovski, V.; and Punera, K. 2013. Towards a robust modeling of temporal interest change patterns for behavioral targeting. In *WWW*, 71–82.

Bansal, T.; Das, M. K.; and Bhattacharyya, C. 2015. Content driven user profiling for comment-worthy recommendations of news and blog articles. In *RecSys*, 195–202.

Blei, D. M., and Lafferty, J. D. 2006. Dynamic topic models. In *ICML*, 113–120.

Cantador, I.; Brusilovsky, P.; and Kuflik, T. 2011. Second workshop on information heterogeneity and fusion in recommender systems (hetrec2011). In *RecSys*, 387–388.

Chen, C.; Zheng, X.; Wang, Y.; Hong, F.; Lin, Z.; et al. 2014. Context-aware collaborative topic regression with social matrix factorization for recommender systems. In *AAAI*, 9–15.

Dai, H.; Wang, Y.; Trivedi, R.; and Song, L. 2016. Recurrent coevolutionary latent feature processes for continuous-time recommendation. In *DLRS*, 29–34.

Du, N.; Wang, Y.; He, N.; Sun, J.; and Song, L. 2015. Time-sensitive recommendation from recurrent user activities. In *NIPS*, 3492–3500.

Duchi, J.; Shalev-Shwartz, S.; Singer, Y.; and Chandra, T. 2008. Efficient projections onto the l 1-ball for learning in high dimensions. In *ICML*, 272–279.

Gad, S.; Javed, W.; Ghani, S.; Elmqvist, N.; Ewing, E. T.; Hampton, K. N.; and Ramakrishnan, N. 2015. Themedelta: Dynamic segmentations over temporal topic models. *IEEE Trans. Vis. Comput. Graph.* 21(5):672–685.

Gao, L.; Wu, J.; Qiao, Z.; Zhou, C.; Yang, H.; and Hu, Y. 2016. Collaborative social group influence for event recommendation. In *CIKM*, 1941–1944.

Koren, Y. 2009. Collaborative filtering with temporal dynamics. In *SIGKDD*, 447–456.

Kurashima, T.; Iwata, T.; Hoshide, T.; Takaya, N.; and Fujimura, K. 2013. Geo topic model: joint modeling of user's activity area and interests for location recommendation. In *WSDM*, 375–384.

Li, B.; Zhu, X.; Li, R.; Zhang, C.; Xue, X.; and Wu, X. 2011. Cross-domain collaborative filtering over time. In *IJCAI*, 2293–2298.

Liu, X. 2015. Modeling users dynamic preference for personalized recommendation. In *IJCAI*, 1785–1791.

Lu, Z.; Dou, Z.; Lian, J.; Xie, X.; and Yang, Q. 2015. Content-based collaborative filtering for news topic recommendation. In *AAAI*, 217–223.

Lu, Z.; Pan, J.; Li, Y.; Jiang, J.; and Yang, Q. 2016. Collaborative evolution for user profiling in recommender systems. In *IJCAI*, 3804–3810.

Neumaier, A., and Schneider, T. 2001. Estimation of parameters and eigenmodes of multivariate autoregressive models. *ACM Transactions on Mathematical Software (TOMS)* 27(1):27–57.

Purushotham, S.; Liu, Y.; and Kuo, C. J. 2012. Collaborative topic regression with social matrix factorization for recommendation systems. In *ICML*.

Salakhutdinov, R., and Mnih, A. 2007. Probabilistic matrix factorization. In *NIPS*, 1257–1264.

Wang, C., and Blei, D. M. 2011. Collaborative topic modeling for recommending scientific articles. In *SIGKDD*, 448–456.

Wang, B., and Klabjan, D. 2016. Temporal topic analysis with endogenous and exogenous processes. In *AAAI*, 3045–3051.

Wang, H.; Chen, B.; and Li, W. 2013. Collaborative topic regression with social regularization for tag recommendation. In *IJCAI*, 2719–2725.

Wang, H.; Wang, N.; and Yeung, D. 2015. Collaborative deep learning for recommender systems. In *SIGKDD*, 1235–1244.

Zhang, W., and Wang, J. 2015. A collective bayesian poisson factorization model for cold-start local event recommendation. In *SIGKDD*, 1455–1464.

Zhang, H.; Kim, G.; and Xing, E. P. 2015. Dynamic topic modeling for monitoring market competition from online text and image data. In *SIGKDD*, 1425–1434.

Zhu, J., and Xing, E. P. 2011. Sparse topical coding. In *UAI*, 831–838.

Zivot, E., and Wang, J. 2006. Vector autoregressive models for multivariate time series. *Modeling Financial Time Series with S-PLUS®* 385–429.