EXPLORE: Explainable Item-Tag Co-recommendation

Xiaolin Zheng[a], Menghan Wang[a,*], Chaochao Chen[a], Yan Wang[b], Zhehao Cheng[a]

[a] ~~the~~ *College of Computer Science, Zhejiang University,Hangzhou 310027, China*
[b] ~~the~~ *Department of Computing, Macquarie University, Sydney, NSW 2109, Australia*

Abstract

~~Due to the popularization of social tagging systems,~~ tag-based recommendation has become increasingly important in recent years. The related studies can be categorized into item recommendation and tag recommendation based on the objective of recommendation. In both categories, most existing recommendation approaches focus on improving the prediction accuracy. However, they ignore that the explanations of recommendations also greatly affect the decision-making of users~~, which tell users why they might like the recommended items~~. In social tagging systems, tags not only behave as the auxiliary information of items but also show the implicit preferences of users. Therefore, they can be used for both improving the prediction accuracy and providing explanations to the recommended items. Meanwhile, items and tags have interrelation and mutual effects. Focusing on either item recommendation or tag recommendation may miss some information and can only achieve marginal gains. Fusion of the two types of recommendations would improve the performance of both item and tag recommendation. Based on the above ideas, in this paper, we propose an EXPLainable item-tag CO-REcommendation framework (EXPLORE) that jointly recommends items and the corresponding tags. Different from conventional recommendations that utilize a single source of content, EXPLORE takes advantage~~s~~ of user's interests, item's contents, and item's tags. The ex-

*Corresponding author

Email addresses: `xlzheng@zju.edu.cn` (Xiaolin Zheng), `wangmengh@zju.edu.cn` (Menghan Wang), `chaochao.ccc@antfin.com` (Chaochao Chen), `yan.wang@mq.edu.au` (Yan Wang), `chengzhehaochn@gmail.com` (Zhehao Cheng)

Preprint submitted to Journal of LaTeX Templates *April 29, 2018*

periments conducted on three real-world data sets demonstrate that EXPLORE outperforms the state-of-the-art methods. More importantly, the recommended tags could provide explanations to the recommended items, thus making the recommendation results explainable.

Keywords: Collaborative filtering, Recommender systems, Topic models, Co-recommendation.

2010 MSC: 00-01, 99-00

---

1. Introduction

Recommender systems have been studied and deployed in various domains. They become indispensable as they help to deal with the information overload problem and provide customized recommendations based on users’ preferences. Collaborative Filtering (CF) is one of the most successful approaches to recommendation Su et al. (2009), Huang et al. (2007). The core idea of CF is that users with similar tastes share similar rating distributions toward the same items. Popular methods combine CF with auxiliary information to make recommendations.

Recently, social tagging systems (e.g., Delicious, CiteULike, and Flickr) have become prevalent and accumulated abundant tags. These tags contain rich information and provide effective ways for users to organize, manage, share, and search various kinds of items Chen et al. (2016). Researchers have exploited tags to improve recommendations in various methods. Based on the objectives of recommendations, we can categorize the existing studies into two groups: (1) tag-based item recommendation, and (2) tag recommendation. Tag-based item recommendations use tags as auxiliary information to improve the recommendation accuracy. Some researchers used tags to calculate the similarity of users Zhen et al. (2009) or items Zhou et al. (2009), and applied these similarities into the CF procedure. Some other researchers mined tags’ semantic meanings to model users’ interests and item’s features Chen et al. (2016), Tso-Sutter et al. (2008), Zhang et al. (2009). These studies confirmed the abundance and semantic meanings of tags could help improve the performance of recommender systems. Tag recommendation refers to the automated process of suggesting useful and informative tags to an emerging object based on historical information Song et al. (2011). Researchers explored tags' co-occurrences Sigurbjörnsson & Van Zwol (2008), the interactions between items and tags Wang et al. (2013), Lops et al. (2013), Lipczak et al. (2009), Fang et al. (2015), and the semantic meanings of tags Gupta et al. (2010), Krestel et al. (2009) to recommend tags. For both item recommendation and tag recommendation, CF-based techniques have been well studied showing promising results.

However, most existing studies only focus on improving the accuracy of item or tag recommendations. In reality, a social tagging system need both item and tag recommendation. When viewing them in a unified perspective, we can utilize more coherent features and mine the relations between two recommendations. We can care not only the accuracy but also other factors that influence recommendation (e.g., recommendation novelty, explanation, metric design). Besides the accuracy, in this paper we mainly consider two issues that have been overlooked in single recommendation scenarios.

One issue is that recommendations lack explainability. The explainability of recommendations reveals why users might like the items that a recommender system has recommended, helping users make ~~more~~ appropriate decisions. For example, as depicted in Fig.1, we recommend two papers about CF to users. The tags serve as the explanations to the papers. Two users with different interests look through the recommended list to decide which paper to read. Alice is new to recommender system and wants to learn about what CF is. She finds the first paper *"Item-Based Collaborative Filtering Recommendation Algorithms"* is explained with *"Classic"* and *"Beginner"*, then she decides to read this paper. Bob is an engineer in distributed system and has some knowledge about CF. He wants to implement CF in a distributed style and ~~he~~ chooses the second paper *"Fast Item-based Collaborative Filtering"* when he sees the explanations are *"Parallel"*, *"Clustering"*, and *"Hashing"*. Both Alice and Bob make the right decisions with the tags serving as the explanations to the rec-

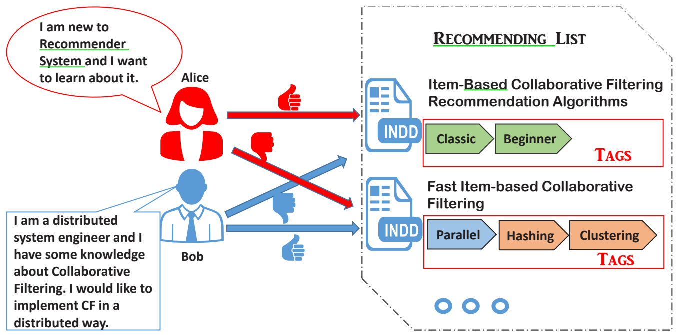


Figure 1: An example of recommending papers with explainability. The tags serve as explanations. Two people are deciding which paper to read based on the paper titles and paper explanations.

ommended papers. In contrast, one may not choose the right paper without any explanations. Because the titles of two papers are too similar and users may take time to tell the difference between them~~, especially for those who lack preliminary knowledge~~. Thus, the explanations play an important role in ~~users'~~ decision making. Compared to other forms of explanations (e.g., keywords of items Billsus & Pazzani (1999), similar users' choices Bilgic & Mooney (2005), and items' features Tintarev (2007)), tags contain both factual and subjective information that are more friendly and acceptable to users. However, tags are human annotated and unevenly distributed, not every item has sufficient tags for explanation.

The other issue is the metric design for tag recommendation. A proper metric is vital to evaluate the effectiveness of a recommender system. Existing works use all the items to compute the evaluation metrics for tag recommendation (e.g., precision, recall), assuming users will look through each item with the same probability and treating items equally in the evaluation procedure. While in real scenarios, items have different exposure to users. Some popular items are frequently searched or recommended, and the accuracy of these~~s~~ items' recommended tags is important for a social tagging system to improve user experience. Other unpopular items draw less attention~~s~~ from users and the recommended tags of these items have less influence on user experience. Ignoring this issue may lead to biased evaluation. An ideal item subset could be based on item popularity or item exposure to users, which, however, is hard to compute. A reasonable alternative is needed.

We come up with the idea of item-tag co-recommendation to address the above two issues. It means the recommender system recommends an item and the corresponding tags to users simultaneously. For the *explainability issue*, we use recommend tags so every item has the recommended tags as explanations. In this way the problem of unevenly distributed tags are alleviated and then the item-tag co-recommendation can provide better explanation to recommended items. For the *metric issue*, we design a more meaningful metric when the recommended items and recommended tags are available at the same time. The recommended items can be viewed as the exposures to users. We test the tag recommendation performance by only using users' recommended items instead of the whole set of items. It is a novel metric and provides a new perspective to observe the effectiveness of a recommender system. We further realize this idea by a novel model combining the training phases of item recommendation and tag recommendation, and then the items and their corresponding tags could be ~~recommended simultaneously.~~

Moreover, the item-tag co-recommendation has two additional advantages: (1) The training phases are unified under the same framework. It indicates that item-tag co-recommendation can model the interrelation and mutual effects by simultaneously updating the common factors used in item recommendation and tag recommendation. As all the parameters are tuned together, we can also explore the tradeoff between item recommendation and tag recommendation; (2) More different aspects of auxiliary information can be exploited to improve the performance. The existing item recommendation or tag recommendation methods often utilize the content from a single source (e.g., user profiles, product descriptions, reviews, or tags) to capture latent factors, which exploits one aspect of items (or users). In our item-tag co-recommendation, multiple aspects of content information are involved and we can utilize all of them rather than one of them to capture the latent factors of items. Here we leverage ~~user's~~

implicit feedback, item's descriptions, and item's tags to capture items more comprehensively. This method also helps alleviate the sparsity problem, which means that the recorded data is extremely sparse compare to the user-product matrix.

In this paper, we propose a novel unified framework named ~~“EXPLORE” (short for ”~~EXPLainable item-tag CO-REcommendation”) to integrate item recommendation and tag recommendation for providing explanations to recommendation results, and boosting prediction performances both on items and tags. The main contributions are listed as follows:

- We propose a novel co-recommendation framework that jointly recommends items and corresponding tags simultaneously. To the best of our knowledge, it is the first work to use the content CF-based techniques to do co-recommendation.
- The recommended tags serve as explanations to the recommended items, making the recommendation more convincing and interpretable. We also introduce a new metric to evaluate the tag recommendation.
- We capture the latent factors of an item by considering user's implicit feedback, item's content, and the tags people annotated to. Unlike utilizing a single source of content in most existing studies, we take advantage of multi-aspects of content to capture items more comprehensively.
- Experiments on real-world datasets show that EXPLORE achieves higher performance in both item and tag recommendation than the state-of-the-art methods. The co-recommendation proves to be a mutual promotion.

The remainder of this paper is organized as follows. In Section 2, we introduce the related work including item recommendation, tag recommendation, topic-based CF techniques, and recommendation explainability. In Section 3 we describe the details of our EXPLORE framework. Section 4 shows the experiments and discussions. Section 5 discuss~~es~~ the conclusions and future works.

2. Related Work

In this section, we review the related work of this paper in three groups, including (1) CF-based recommendation, (2) recommendation explanability, and (3) ~~the~~ Collaborative Topic Regression.

2.1. CF-based Recommendation

CF-based Recommendations have proved their effectiveness and become a trend in the recommendation area. Here we focus on the hybrid approaches integrated with auxiliary information in the setting of social tagging systems. We briefly introduce the state-of-the-art hybrid models for item recommendation and tag recommendation separately.

Item recommendation. Researchers have exploited various auxiliary information to alleviate the cold start problem and improve the prediction performance of item recommendation. Trust-based approaches Andersen et al. (2008),Jamali & Ester (2009) incorporate trust relationships between users to mine the similarity among users rather than the rating similarity calculated in pure CF approaches. Social network based approaches Purushotham et al. (2012), Chen et al. (2014) studied the influence of users' social circle and capture the similarity by groups. Content-based approaches enhance user and item representations by incorporating additional user or item information such as user profiles, item descriptions, customer reviews Pazzani & Billsus (2007), Wang & Blei (2011), Hu et al. (2015). The prevalent approach is to extract topic information from content and apply them to CF techniques in proper ways, which we will discuss later. In social tagging systems, tags are used as auxiliary information to improve item recommendation. Zhou et al. Zhou et al. (2009) proposed a unified probabilistic matrix factorization that connects user-item matrix, user-tag matrix, and item-tag matrix. It doesn't utilize the content information and the complexity is linear with respect to the number of observations in the three sparse matrices. Karen et al. Tso-Sutter et al. (2008) extended the user-item matrix by including user tags as items and item tags as users, i.e., the fusion of users and items. The fusion based prediction was done by a combination of standard user and item based CF. Zhen et al. Zhen et al. (2009) proposed a framework named tag informed collaborative filtering (TagiCoFi) to seamlessly integrate tagging information into the CF procedure. TagiCoFi employs the user similarities defined based on the tagging information to regularize the MF procedure. Chen et al. Chen et al. (2016) used tags as a bridge to capture the implicit semantic correlation between users and items in social tagging systems. The difference between EXPLORE and these content CF-based methods is that EXPLORE exploits multiple aspects of content to capture item's latent factors. Recently, some researchers Wang et al. (2015), Li et al. (2015), Covington et al. (2016) utilized deep learning techniques to learn the item features from text and showed promising performances.

Tag recommendation. There has been several works that use collaborative filtering to recommend tags. Their difference lies in accounting similarities between users, resources, and tags. Some researchers explored the co-occurrence of tags Sigurbjörnsson & Van Zwol (2008), k best neighbours of the item's tags Marinho & Schmidt-Thieme (2008), TF-IDF Xu et al. (2006), and Latent Dirichlet Allocation Krestel et al. (2009) to calculate the similarities. Recent hybrid methods ~~showed promising performance~~ Song et al. (2011), Lops et al. (2013), Rendle & Schmidt-Thieme (2010), Wang et al. (2013)~~, which combined both~~ item-tag matrix and item content information for recommendation. Rendle Rendle & Schmidt-Thieme (2010) presented a factorization model that learned with an adaption of the Bayesian personalized ranking criterion. It works by modeling the pairwise interactions between users, items and tags. Wang et al. Wang et al. (2013) introduced a hierarchical Bayesian model combing collaborative topic regression and social regularization to recommend tags to items. The model integrates the item-tag matrix, item content information, and social networks between items. Song et al. Song et al. (2011) proposed a document-centered model for tag recommendation. It works by finding the most representative documents within the data collections and advocates a sparse multi-class Gaussian process classifier for efficient document classification. Documents are first classified into topic clusters and then the most relevant tags ~~of~~ are selected as recommended tags.

2.2. Recommendation Explainability

Recommendation explainability can help users make more accurate decisions, improve user acceptance of recommendations, and enhance trustfulness in the recommender system Vig et al. (2009). The common solution to explanation is to find an explainable intermediary entity bridging users and recommended items. An intermediary entity is needed because the direct relationship between user and item is always unknown or implicit. User-based intermediary entity uses similar users' choices to explain the recommendation Herlocker et al. (2000), Bilgic & Mooney (2005). Feature-based intermediary entity uses predefined categories (e.g., genre, director, cast) and keywords of items (e.g., books, articles, websites)Tintarev (2007), Billsus & Pazzani (1999), Mooney & Roy (2000). However it is limited as some types of items such as music or pictures may not have available textual content. In recent years tags have been exploited to provide explanations for recommendations Vig et al. (2009). As a source of user generated contents, tags are more relevant to items and more friendly and preferred by users.

2.3. Collaborative Topic Regression

Collaborative Topic Regression (CTR) Wang & Blei (2011) is the most related work to our paper and is the preliminary work of our model. CTR is proposed to recommend documents to users by seamlessly integrating ~~both~~ feedback matrix and item (document) content information into the same model. CTR provides an interpretable latent structure for users and items, and it can form recommendations about both existing and newly published articles. CTR assumes there are K topics $\beta = \beta_{1:k}$. Then the generative process of CTR is as follows:

1. For each user i, draw a user latent vector $u_i \sim \mathcal{N}(0,\lambda_u^{-1} I_K)$.
2. For each item j:

(a) Choose $\theta_j \sim \text{Dirichlet}(\alpha)$.

(b) Draw the topic proportion $\epsilon_j \sim \mathcal{N}(0,\lambda_v^{-1} I_K)$ and set the item latent vector as $v_j = \epsilon_j + \theta_j$.

(c) For each word w_{jn}:

  (a) Draw the topic assignment $z_{jn} \sim \text{Mult}(\theta)$,

  (b) Draw the word $w_{jn} \sim \text{Mult}(\beta_{z_{jn}})$,

3. For each user-item pair (i,j), draw the rating: $r_{ij} \sim \mathcal{N}(u_i^T\ v_j,\ c_{ij}^{-1})$.

CTR introduces an item latent offset ϵ_j between the topic proportion u_j in LDA and the item latent vector v_j in CF. The offset can be explained by the gap between what the article is about (represented by u_j) and what the users think of it (represented by v_j). CTR achieves ~~a~~ high prediction performance and shows a promising direction of exploiting text information in recommendation. CTR has been widely used as baselines in recent recommendation studies and many recommendation models are the extensions of it.

3. EXPLORE: An Explainable Item-Tag Co-recommendation framework

We now focus on describing the details of our proposed framework - EXPLORE. We first formulate the problem of item-tag co-recommendation and then introduce the model formulation of EXPLORE by explicating both the parameter estimation algorithm and prediction method. Some important notations are explained in Table 1.

3.1. Problem Definition

Assume we have a set of users $U = \{u_i\}_{i=1}^{I}$, a set of items $V = \{v_j\}_{j=1}^{J}$, a set of tags $T = \{t_r\}_{r=1}^{R}$, and two rating matrices R^I and R^T. $R^I \in \mathbb{R}^{U\times I}$ is a user-item rating matrix where the element $r_{I_{ij}}$ refers to the rating that user i gives to item j, $R^T \in \mathbb{R}^{I\times R}$ is an item-tag rating matrix where the element $r_{T_{jr}}$ refers to the tag r that is given to item j. $r_{I_{ij}} > 0$ means user i is interested

Table 1: Notation

Symbol	**Description**
U, V, T	latent factors for users, items, tags
C	a rating confidence matrix
$r_{I_{ij}}$	the rating that user u_i gives to v_j
$r_{T_{jr}}$	the tag t_r that is given to item I_j
K	number of latent dimensions/topics
α_u, α_v	Dirichlet parameters
θ_i	user interest proportions of u_i
θ_j	item topic proportions of v_j
$\lambda_u, \lambda_v, \lambda_t$	regularization parameters for U, V, T
λ_{uv}	the contribution of implicit preference
λ_{vt}	contribution of tag information to
w_{in_u}	n^{th} word of interests of user u_i
w_{jn_v}	n^{th} word of content of item v_j
z_{in_u}	topic for the n^{th} word of interests of user u_i
z_{jn_v}	topic for the n^{th} word of interests of user v_j
$\beta_{z_{in_u}}, \beta_{z_{jn_v}}$	word probability of w_{in_u}, w_{jn_v}

in item j, and $r_{I_{ij}} = 0$ means unknown as user i may dislike item j or may not know j.

For item recommendation, we aim to predict the unknown rating $r_{I_{ij}}$ from a user i to an item j. For tag item recommendation, we aim to predict the unknown rating $r_{T_{jr}}$ from an item j to a tag r. After calculating all the unknown ratings we choose the tags with top ratings to recommend. For the item-tag co-recommendation in our method EXPLORE, we aim to recommend a user i a list of items with top ratings. And for each of the recommended items, we recommend a list of tags with top ratings. Items and tags are recommended simultaneously, and the tags serve as the explanations to the items. The training procedures are unified. The key point of EXPLORE is how to combine item recommendation and tag recommendation in the training procedures for a better performance.

3.2. Model Formulation

The basic idea to combine item recommendation and tag recommendation is to set the item latent factor V as a shared component. In item recommendation we predict a rating user i gives to item j by $r_{I_{ij}} = u_i^T v_j$, and in tag recommendation we predict a rating of a tag r with a item j by $r_{T_{jr}} = v_j^T t_r$. In a unified model we use one latent factor v_j to represent an item for both recommendations. This is a simple combination of two collaborative filtering models that only utilize rating information.

Further, we incorporate text information into the collaborative filtering models. We run Latent Dirichlet Allocation (LDA) Blei et al. (2003) on text information of item's content and user's interests, and get topic proportion θ_i for item and topic proportion θ_j for users separately. The topic proportions are fixed vectors and are well fitted to latent factor models. In this paper we apply the same strategy in CTR Wang & Blei (2011): initialize latent factors with topic proportions, and add an offset variable to balance the effectiveness between text information and rating information. To utilize more aspects of auxiliary information, we introduce two arrows to connect U and T to V. $U \rightarrow V$ indicates that we utilize the interests of users who rated the item to capture the item's latent factors. These interests are implicit feedbacks of users reveal~~ing~~ the implicit characteristics of items that attract certain group of users, and $T \rightarrow V$ indicates the utilization of tags to capture item's latent factors. These tags are explicit feedbacks of users showing the user experiences and the supplementary description of items. Note the θ_j shows natural characteristics of items. The

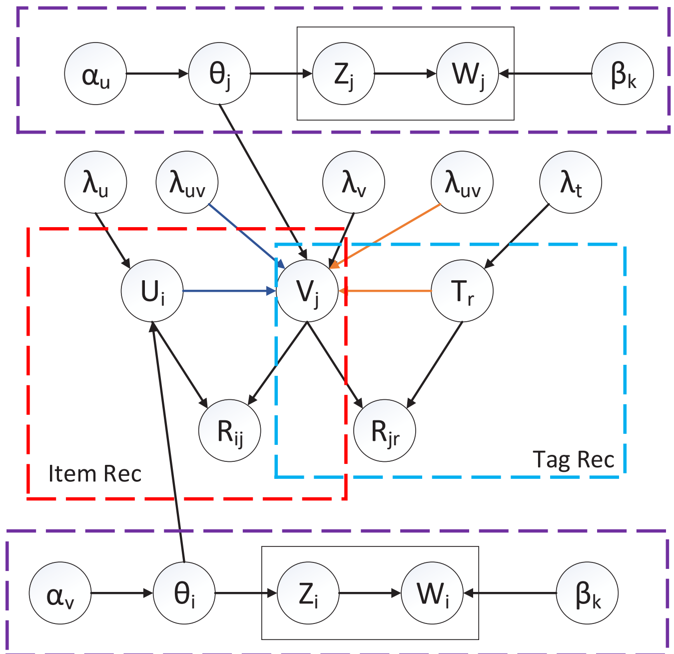


Figure 2: The graphical representation of EXPLORE. The red dashed rectangle is the item recommendation part, the blue dashed rectangle is the tag recommendation part, and the purple dashed rectangles are the LDA part.

underlying intuition is that all three aspects of the auxiliary information could contribute to constructing a more comprehensive latent factor of items.

To better describe our proposed framework, we show the graphical representation of EXPLORE in Fig. 2. The red dashed rectangle is the item recommendation part, the blue dashed rectangle is the tag recommendation part, and the purple dashed rectangles are the LDA part. The generative process of EXPLORE is as follows:

1. For each user u_i,
    (a) Draw the user interest proportions $\theta_i \sim \text{Dirichlet}(\alpha_u)$;
    (b) Draw the user latent offset $\epsilon_i \sim \mathcal{N}(0, \lambda_u^{-1} I_K)$ and set the user latent vector as $u_i = \epsilon_i + \theta_i$;
    (c) For each word w_{in_u} in the interests of user u_i,
        i. Draw the topic assignment $z_{in_u} \sim \text{Mult}(\theta_i)$;

ii. Draw the word $w_{in_u} \sim \text{Mult}(\beta_{z_{in_u}})$;

2. Draw the tag latent vector for each tag t_r, $t_r \sim \mathcal{N}(0,\, \lambda_t^{-1} I_K)$;
3. For each item v_j,
   (a) Draw the item topic proportions $\theta_j \sim \text{Dirichlet}(\alpha_v)$;
   (b) Draw the item latent vector as

$$lv_j \sim \mathcal{N}(\theta_j, \lambda_v^{-1} I_K) \times \prod_i I_{ij}^R \mathcal{N}(u_i, \lambda_{uv}^{-1} I_K) \times \prod_r I_{jr}^R \mathcal{N}(t_r, \lambda_{vt}^{-1} I_K).$$

   (c) For each word w_{jn_v} in the content of item v_j,
      i. Draw the topic assignment $z_{jn_v} \sim \text{Mult}(\theta_j)$;
      ii. Draw the word $w_{jn_v} \sim \text{Mult}(\beta_{z_{jn_v}})$;
4. For each user-item pair (i, j), draw the item rating:

$$r_{I_{ij}} \sim \mathcal{N}(u_i^T v_j, c_{I_{ij}}^{-1}); \tag{1}$$

5. For each item-tag pair (j, k), draw the tag rating:

$$r_{T_{jr}} \sim \mathcal{N}(v_j^T t_r, c_{T_{jr}}^{-1}). \tag{2}$$

In the above generative process, $\mathcal{N}(x|\mu, \sigma^2)$ is a Gaussian distribution with a mean μ and a variance σ^2, and I_K is an identity matrix with K rows and K columns. I_{ij}^R is an indicator function, the value of which equals ~~to~~ 1 if user u_i rated item v_j, and 0 otherwise. I_{jr}^R is an indicator function the value of which equal to 1 if tag t_r is attached to item v_j, and 0 otherwise. C is a rating confidence matrix with element $c_{i,j}$ denoting the rating confidence, which is similarly defined in Wang & Blei (2011).

The parameter λ_u balances the contribution of user semantic information provided by user's interests and rating information to the model performance. Similarly, the parameter λ_v balances the contribution of item content to items' latent factors V_j. The parameter λ_{uv} balances the contribution of implicit preference of users to latent factors of items on model performance, and the parameter λ_{vt} balances the contribution of tag information to latent factors of items.

3.3. Parameter Estimation

Given parameters λ_u, λ_v, λ_t, λ_{uv}, λ_{vt}, β_u, and β_v, our goal is to compute the maximum a posterior (MAP) estimation of U_j, V_j, T_j, θ_i, and θ_j. The full posterior of U, V, T, θ_u, and θ_v is

$$\begin{aligned} p(U,V,T,\theta_u,\theta_v|R_I,R_T,C,\lambda_u,\lambda_v,\lambda_t,\lambda_{uv},\lambda_{vt},\beta_u,\beta_v,\alpha) \\ \propto p(R_I|U,V,T,C)\times p(R_T|U,V,T,C) \\ \times p(U|\theta_u,\lambda_u)\times p(V|U,T,\theta_v,\lambda_{uv},\lambda_v,\lambda_{vt}) \\ \times p(T|\lambda_t)\times p(\theta_u|\beta_u,\alpha)\times p(\theta_v|\beta_v,\alpha) \end{aligned} \tag{3}$$

However, computing the posterior directly is intractable. The Maximization of the posterior is equivalent to maximizing the complete log likelihood of U, V, T, θ_u, θ_v, R_I, and R_T given λ_u, λ_v, λ_t, λ_{uv}, λ_{vt}, β_u, and β_v.

$$\begin{aligned} \mathcal{L} = -\frac{\lambda_u}{2}\sum_i (u_i-\theta_i)^T(u_i-\theta_i) \\ -\frac{\lambda_v}{2}\sum_j (v_j-\theta_j)^T(v_j-\theta_j) - \frac{\lambda_t}{2}\sum_r t_r^T t_r \\ +\sum_i\sum_{n_u} log(\sum_k \theta_{ik}\beta_{k,w_{in_u}}) \\ +\sum_j\sum_{n_v} log(\sum_k \theta_{jk}\beta_{k,w_{jn_v}}) \\ -\frac{\lambda_{uv}}{2} I_{ij}^R \sum_i (u_i-v_j)^T(u_i-v_j) \\ -\frac{\lambda_{vt}}{2} I_{rj}^T \sum_i (t_r-v_j)^T(t_r-v_j) \\ -\sum_{ij}\frac{c_{ij}^I}{2}(r_{ij}^I-u_iv_j)^2 - \sum_{rj}\frac{c_{ij}^T}{2}(r_{rj}^T-t_rv_j)^2. \end{aligned} \tag{4}$$

We omit a constant and set the Dirichlet priors $\alpha_u = \alpha_u = 1$. Since the posteriori is not convex and hard to find its global optima directly. Inspired by the parameter estimation procedure in CTR, we optimize the function by using the coordinate ascent algorithm, which iteratively optimizes the collaborative filtering variables u_i, v_j, t_k and the topic proportion θ_i, θ_j. For u_i, v_j, t_k, given the current estimate of θ_i, θ_j, we take the gradient of $\mathcal{L}$ in Equation (4) with respect to u_i, v_j, and t_k and set it to zero, leading to:

$$\begin{aligned}
u_i \leftarrow (VC_i^I V^T + \lambda_u I_k + \lambda_{uv} \sum_j I_{ij}^R I_k)^{-1} \\
(VC_i^I R_i^I + \lambda_u \theta_i + \lambda_{uv} \sum_j I_{ij}^R v_j) \\
t_r \leftarrow (VC_r^t V^T + \lambda_t I_k + \lambda_{vt} \sum_j I_{rj}^R I_k)^{-1} \\
(VC_r^t R_r^t + \lambda_u \theta_i + \lambda_{uv} \sum_j I_{ij}^R v_j) \\
v_j \leftarrow (UC_j^I U^T + TC_j^t T^T + \lambda_v I_k + \lambda_{uv} \sum_i I_{ij}^R I_k \\
+\lambda_{vt} \sum_r I_{rj}^t I_k)^{-1} \\
(UC_j^I R_j^I + UC_j^t R_j^t + \lambda_v \theta_j + \lambda_{uv} \sum_i I_{ij}^R v_j \\
+\lambda_{vt} \sum_r I_{rj}^t T_r),
\end{aligned} \tag{5}$$

where C_i^I is a diagonal matrix with $c_{I_{ij}}, j = 1, ..., J$ as its diagonal elements, and $R_i^I = (r_{ij}^I)_{j=1}^J$ for user u_i. The C_r^t is a diagonal matrix with $c_{T_{rj}}, j = 1, ..., J$ as its diagonal elements, and $R_r^t = (r_{rj}^T)_{j=1}^J$ for tag t_r. For item j, $C_j^I, C_j^t, R_j^I, R_j^t$ are similarly defined.

Given U, V, and T, we now update the topic proportions θ_i and θ_j. For θ_i, we first define $q(z_{jn} = k) = \phi_{jnk}$, then separate the users that contain θ_i and apply Jensen's inequality,

$$\begin{aligned}
\mathcal{L}(\theta_i) \quad & \geq -\tfrac{\lambda_u}{2}(u_i - \theta_i)^T(u_i - \theta_i) \\
& + \textstyle\sum_{n_u} \sum_k \Phi_{in_u k}(log\theta_{ik}\beta_{k,w_{in_u}} - log\Phi_{in_u k}) \\
& = \mathcal{L}(\theta_i, \phi_i).
\end{aligned} \tag{6}$$

Here, $\Phi_i = \Phi_{in_u K_{n_u=1,k=1}}^{N_u \times K}$. $\mathcal{L}(\theta_i, \phi_i)$ is the tight lower bound of $\mathcal{L}(\theta_i)$ and we use projection gradient Bertsekas (1997) to optimize θ_i. For θ_j, it is updated in a similar way. The optimal ϕ_{ink} and ϕ_{jnk} are

$$\Phi_{in_uk} \propto log\theta_{ik}\beta_{k,w_{in_u}},$$
$$\Phi_{jn_vk} \propto log\theta_{jk}\beta_{k,w_{jn_v}}. \tag{7}$$

As for the parameter β, we update with the same M-step for topics in LDA.

$$Ll\beta_{kw} \propto \sum_j \sum_n \phi_{jnk} 1[w_{jn} = w]. \tag{8}$$

3.4. Rating Prediction

After the parameters U, V, T, θ_u, θ_v,β_u, and β_v are learned, our framework can be used for item recommendation and tag recommendation, both of which include in-matrix and out-matrix predictions.

Similar to Wang & Blei (2011), for in-matrix predictions, we use the following equations:

$$r^*_{I_{ij}} \approx (u^*_i)^T v^*_j, \tag{9}$$

$$r^*_{T_{jr}} \approx (v^*_j)^T t^*_r. \tag{10}$$

For out-matrix predictions, we use following equations:

$$r^*_{I_{ij}} \approx (u^*_i)^T \theta^*_j, \tag{11}$$

$$r^*_{T_{ij}} \approx (\theta^*_j)^T t^*_r. \tag{12}$$

3.5. Discussion

In this section, we will discuss several detailed issues for our proposed approach.

From the model in the Fig. 2 we can see the latent variable V is involved with latent variables U, T, and item topic distributions θ_j, which indicates that we utilize users' implicit feedbacks, item's tags, and item's contents to capture item's latent factors. One benefit of this approach is that we can balance the weights of users' implicit feedbacks and tags with two confidence parameters λ_{uv} and λ_{vt}. If we put all the three aspects of sources ~~together~~ to generate item's latent factors, some problems arise when the quantities of different sources are uneven. The source with a small quantity will be overwhelmed by the source with a large quantity, so the final latent factors will be mainly influenced by the source with the largest quantity. Our approach ensures three aspects of the auxiliary information will be used to capture latent factors with settable proportions regardless of the quantities distribution of three sources.

Another detail lies in the parameter estimation phrase. In every loop of parameter~~s~~ update, the v_j is affected by the U and T (Note they are not multiplied by λ_{uv} and λ_{vt} and different from what described in last paragraph). While in a single item recommendation or tag recommendation, the v_j is only affected by U or T. In our framework the training phases are unified. It indicates our framework can model the interrelation and mutual effects between item recommendations and tag recommendations by simultaneously updating the common factors v_j with U and T. In addition, as the parameters are tuned together, we can explore the tradeoff between item recommendations and tag recommendations.

4. Experiments

We conducted comprehensive experiments on three datasets and compared EXPLORE with several baseline algorithms to show the superiority of our framework. The questions we aim to answer are:

1. How does EXPLORE outperform the baseline methods~~,~~ in terms of item recommendation and tag recommendation ~~respectively~~?
2. How is the prediction performance affected by the parameters? What roles do the two confidence parameters λ_{uv} and λ_{vt} play in recommendation?
3. To what extent do the recommended tags provide explanations to the recommended items?

4.1. Datasets

Since our proposed framework exploits ~~the~~ user interests (implicit feedback~~s~~), item content, and tag information for recommendations, we select real well-known datasets that meet these conditions. We use three real-world datasets, two ~~of which are~~ from *CiteULike*[1]~~,~~ and the third ~~one is~~ from *BibSonomy*[2]. These datasets contain user-item ratings, item-tag ratings, item content, and user interests. The content information of items includes titles and abstracts of articles and we process them with the same procedure in Wang & Blei (2011). We remove the stop-words and choose the top 8000 distinct words according to the *tf-idf* values. User interests are the collection of keywords of the articles a user likes, and they are preprocessed in a similar way. To reduce data noise and alleviate the data sparsity problem, we use p-cores Jäschke et al. (2007) to remove infrequent tags, which means the tags we used in experiments occur at least p times. Below is the description of the datasets, some statistical data are shown in Table 2, and the distributions of these entities are shown in Figs. 3-5.

CiteULike is a website that allows researchers to create their own reference libraries for the articles they are interested in and share them with other researchers. This has opened the door to applying recommendation methods Koren et al. (2009) as a third way to help researchers find papers that fit their needs. We used two real datasets from CiteULike named *citeulike-a* and *citeulike-t*. The dataset *citeulike-a* is originally from Wang & Blei (2011), the corresponding tag information is collected by Wang et al. Wang et al. (2013). And the dataset *citeulike-t* is collected by ~~Wang et al.~~ Wang et al. (2013). Note the two datasets don't contain any keywords of articles, which we believe are more concise to capture users' interests. We crawled the keywords of these articles from Baidu Xueshu[3] (a web search engine that indexes the full text or metadata of scholarly literature~~s~~) and used them as the context of interest

[1]http://www.citeulike.org/
[2]http://www.bibsonomy.org/
[3]http://xueshu.baidu.com/

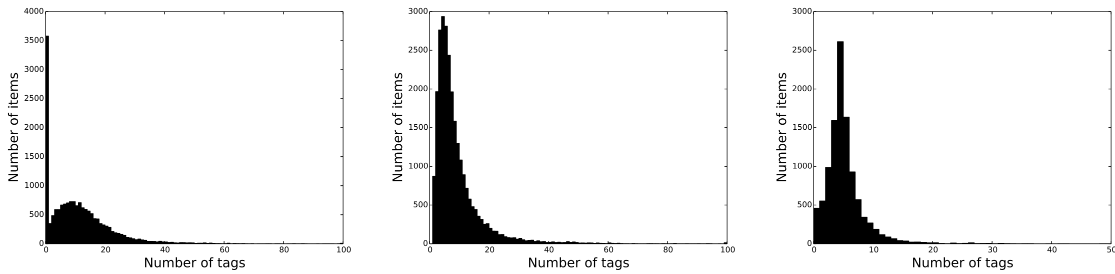
Table 2: Datasets details

	Citeulike-a	**Citeulike-t**	**Bibsonomy**
#users	5551	7947	1047
#items	16980	25975	10662
#tags	7386	8311	4617
#user interests	4951	9871	5910
#rating density	0.217%	0.065%	1.057%
#tagging density	0.145%	0.092%	0.128%
avg #items per user	36.93	16.97	112.66
avg #tags per item	24.60	27.12	4.82
avg #interests	132.01	48.00	43.71

topics. For the p-cores, we set $p = 5$ for both dataset *citeulike-a* and dataset *citeulike-t*.

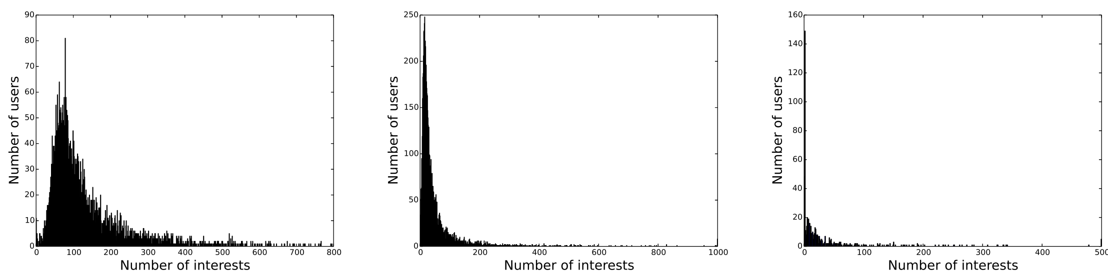
BibSonomy is a social resource sharing system~~s~~ on which users can categorize and archive ~~both~~ bookmarks and literature referencesBenz et al. (2010). The data we used were gathered and made available online by the system administrators[4]. Unlike CiteUlike that focuses on academic papers, BibSonomy supports more literatures like books and publications. Similar to data preparation of Citeulike, we extracted the literature references and crawled corresponding abstracts and keywords from Baidu Xueshu. For the p-cores, we set $p = 3$ as the tag sparsity problem of dataset *BibSonomy* is more severe than that of dataset *citeulike-a* and dataset *citeulike-t*, which is shown in Fig. 3.

[4]Knowledge and Data Engineering Group, University of Kassel: Benchmark Folksonomy Data from BibSonomy, version of June 30th, 2007.



(a) Citeulike-a (b) Citeulike-t (c) Bibsonomy

Figure 3: Item count distribution on number of tags.(a) distribution on *CiteULike-a.* (b) distribution on *CiteULike-t.* (c) distribution on *BibSonomy*.



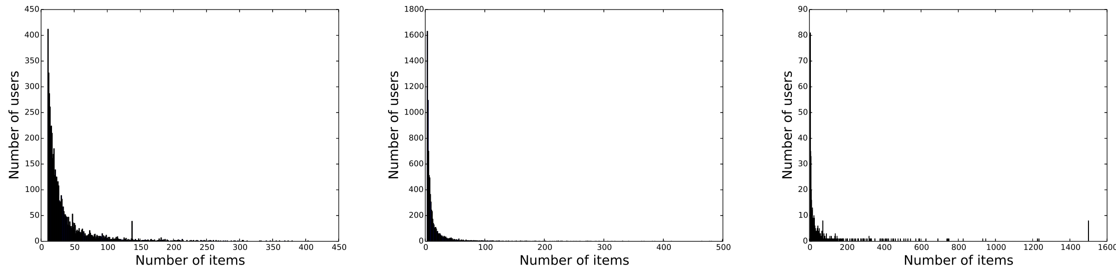
(a) Citeulike-a (b) Citeulike-t (c) Bibsonomy

Figure 4: User count distribution on number of interests.(a) distribution on *CiteULike-a.* (b) distribution on *CiteULike-t.* (c) distribution on *BibSonomy*.

4.2. Evaluation metrics

In this work we adopt accuracy-based metrics and co-recommendation metrics to evaluate our model. Accuracy-based metrics (i.e., recall and precision) are widely used in the recommendation area. And we design a co-recommendation metric to test the performance of co-recommendation compared to recommending items and tags separately.

- **Accuracy-based metrics**: Recall is the proportion of relevant recommended items from the number of relevant items. Recall only considers the positively rated items within the top M recommended items. For each



(a) Citeulike-a (b) Citeulike-t (c) Bibsonomy

Figure 5: User count distribution on number of rated items.(a) distribution on *CiteULike-a*. (b) distribution on *CiteULike-t*. (c) distribution on *BibSonomy*.

user, the definition of $recall_{item}@M$ is

$$recall_{item}@M = \frac{\mid T_{h-item} \cap T^M_{r-item} \mid}{\mid T_{h-item} \mid}, \tag{13}$$

where T_{h-item} is set of items the user likes, T^M_{r-item} is the set of top M items recommended to the user. Similarly the recall of an item is defined as~~:~~

$$recall_{tag}@M = \frac{\mid T_{h-tag} \cap T^M_{r-tag} \mid}{\mid T_{h-tag} \mid}, \tag{14}$$

where T_{h-tag} is the set of the tags that users gave to the item, T^M_{r-tag} is the set of top M tags recommended to the item. Precision is not a proper evaluation here as zero ratings may indicate that a user do not like the items or s/he does not know the items. ~~For tag recommendation,~~ the precision metric also suffers the same problem. For both metrics, the higher the values, the better the performances of recommendation.

- **Co-recommendation metrics**: We propose rec-item tag recall for evaluation of co-recommendation. rec-item tag recall measures the tag recall of the recommended items, rather than all the items. Under this setting, for each user, the rec-item tag recall $recall_{rec-i-t}@M$ is defined as follows:

$$recall_{rec-i-t}@M = \frac{\sum_i recall_{tag}(i)@M}{\mid T^M_{r-item} \mid}, i \in T^M_{r-item} \tag{15}$$

where $recall_{tag}(i)@M$ is the tag recall of item i defined in equation (14), and T^{M}_{r-item} is the set of top M items recommended to the user. For $recall_{rec-i-t}@M$, the higher the value, the better the performance of recommendation.

4.3. Baselines

To evaluate the performance of our proposed EXPLORE framework, we compare it with a variety of baseline algorithms. The baseline algorithms are chosen according to the following criteria: (i) ~~the~~ state-of-the-art algorithms for item prediction, (ii) ~~the~~ state-of-the-art algorithms for tag prediction. In detail, we consider the following algorithms as baselines:

- **PMF**: Probabilistic Matrix Factorization Salakhutdinov & Mnih (2008) ~~(PMF)~~ is a Bayesian generative model which draws user latent feature matrix and item latent feature matrix separately. It is a widely used matrix factorization model for recommendation. The information PMF used is simply the records of ratings. It factorizes the training matrix into two low-rank matrices U and V~~,~~ and recovers the original matrix by UV^T. Here we use PMF model for both item recommendation and tag recommendation.

- **CTR**: Collaborative Topic Regression Wang & Blei (2011) ~~(CTR)~~ combines traditional collaborative filtering with topic modeling for item recommendation. CTR utilizes review~~s~~ information as items~~'~~ content for topic modeling, and it is a widely used baseline model for recent recommender systems that utilize collaborative filtering and content information.

- **CTR+**: this is an improved CTR model, in which the tags as well as reviews are used to form the content of items.

- **TRCF**: this model Chen et al. (2016) uses tags as a bridge to capture the implicit semantic correlation between users and items. It incorporates shared semantic information into matrix factorization and is one of the state-of-the-art models which utilize topic models for recommendation.

- **CDL**: this model Wang et al. (2015) jointly performs deep representation learning for the content information and collaborative filtering for the ratings. It utilizes generalized stacked Denoising Auto-encoders to learn the features.

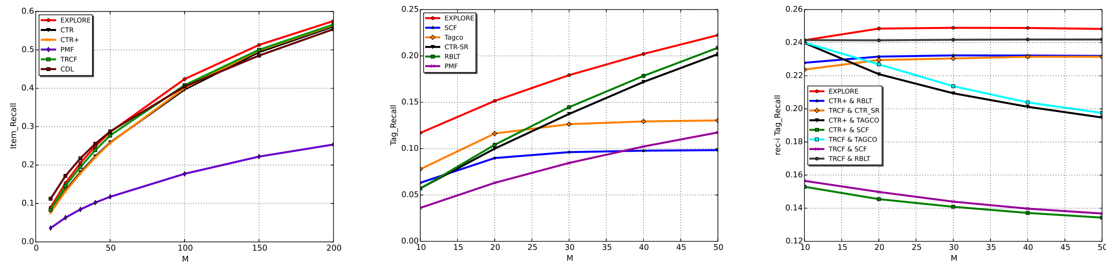
- **TAGCO**: this is a co-occurrence based tag recommendation model Sigurbjörnsson & Van Zwol (2008). This model normalizes the co-occurrence counts with the overall frequency of the tags to measure the quality of the relationship between two tags, then it recommends tags to items based on the scores of tags' qualities.

- **SCF**: this is a similarity-based tag recommendation model Marinho & Schmidt-Thieme (2008). It finds k best neighbours of the item's tags and recommends new tags according to its neighbours' tags.

- **CTR-SR**: this is a tag recommendation model based on CTR with social regularization Wang et al. (2013). CTR-SR incorporates the item-tag matrix, item content, and social networks between items. In this paper we don't consider social factors, and we fix the social matrix as a constant matrix.

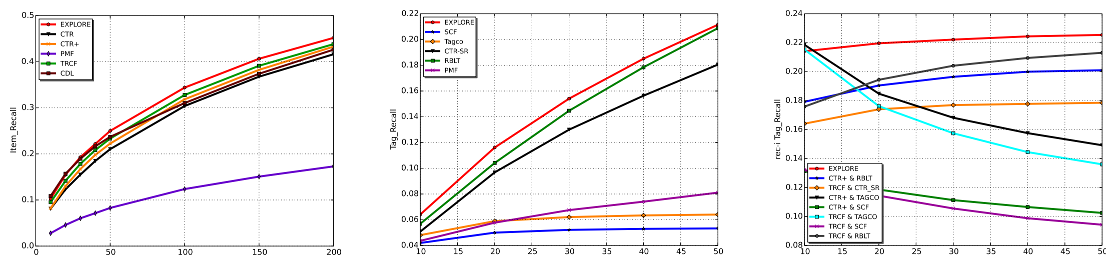
- **RBLT**: Rating-Boosted Latent Topics Tan et al. (2016) ~~(RBLT)~~ boosts raw content with the sentiment orientations of users for topic learning. It is originally proposed to boost review content by rating (e.g., 1-5) times for item recommendation. In this paper, we use it for tag recommendation. we use item content and tags for topic learning and the tags are boosted by occurrence times.

Here we compare the performance of item prediction with the baselines for item recommendation~~,~~ and compare the performance of tag prediction with the baselines for tag recommendation. For the co-recommendation metric, we take a Cartesian product of the experimental results of item recommendation baselines and tag recommendation baselines. and compare them to the performance of our model.



(a) $recall_{item}@M$ (b) $recall_{tag}@M$ (c) $recall_{rec-i-t}@M$

Figure 6: Experimental results on *CiteULike-a* datasets. (a) shows the recall of item recommendation $recall_{item}@M$. (b) shows the recall of tag recommendation $recall_{tag}@M$. (c) shows the rec-item tag recall $recall_{rec-i-t}@M$

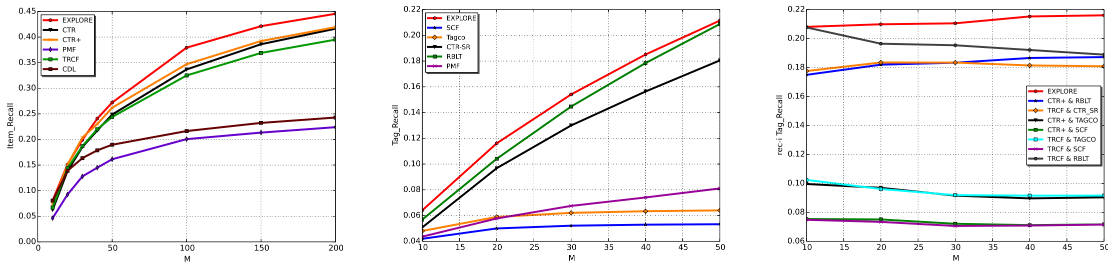


(a) $recall_{item}@M$ (b) $recall_{tag}@M$ (c) $recall_{rec-i-t}@M$

Figure 7: Experimental results on *CiteULike-t* datasets. (a) shows the recall of item recommendation $recall_{item}@M$. (b) shows the recall of tag recommendation $recall_{tag}@M$. (c) shows the rec-item tag recall $recall_{rec-i-t}@M$

4.4. Performance

We evaluate our proposed method on three datasets and compare EXPLORE with the above mentioned baseline algorithms. The results are plotted in Figs. 6-8. We use grid search to find the optimal parameters. For EXPLORE, we set $\lambda_u = 1.0$, $\lambda_v = 10.0$, $\lambda_t = 1.0$, $\lambda_{uv} = 0.1$, $\lambda_{vt} = 1.0$, $K = 200$, $a = 1$, $b = 0.01$, $a_t = 1$, and $b_t = 0.01$. For CTR and CTR+, we set $\lambda_u = 1.0$, $\lambda_v = 100.0$, $K = 200$, $a = 1$, and $b = 0.01$. For TRCF, we set $\lambda_u = 1.0$, $\lambda_v = 10.0$, $\lambda_{uv} = 0.01$, $K = 200$, $a = 1$, and $b = 0.01$. For RBLT, we set $\lambda_u = 1.0$,



(a) $recall_{item}@M$ (b) $recall_{tag}@M$ (c) $recall_{rec-i-t}@M$

Figure 8: Experimental results on *BibSonomy* datasets. (a) shows the recall of item recommendation $recall_{item}@M$. (b) shows the recall of tag recommendation $recall_{tag}@M$. (c) shows the rec-item tag recall $recall_{rec-i-t}@M$

$\lambda_v = 1.0$, $K = 200$. For CDL, we follow the same setting in Wang et al. (2015).

4.4.1. Performance on item recommendation

For item recall, we set $M = 10$, 20, 30, 40, 50, 100, 150, and 200 respectively. From ~~the~~ part (a) of Figs. 6-8, we can find that: (1) PMF always achieves the worst performance, PMF only uses rating information during the matrix factorization procedure, ignoring the rich information in items' content. ~~Meanwhile~~ the sparsity in user-item matrix also accounts for the poor performance of PMF; (2) CTR outperforms PMF in all the cases as PMF doesn't utilize items' content. PMF only uses rating information during the matrix factorization procedure, while CTR utilizes rating information and content information; (3) CTR and CTR+ perform similarly on *CiteUlike-a* and *BibSonomy* while CTR+ performs a little better than CTR on *CiteUlike-t*. The consequences are influenced by tag quality and quantity in the datasets. The statistics shown in Table 2 indicate the items in dataset *CiteUlike-t* have more tags annotated with them than those in other two datasets, which accounts for the different performance of CTR+ among the datasets; (4) TRCF outperforms CTR+ and CTR on three datasets as it utilizes shared semantic information between users and items. But it uses same content to model user and item, only one aspect of information is used.(5)

CDL has a high performance when $M < 30$, but its performance deteriorates compared to other models when M gets larger. There are mainly two reasons. The first is that deep learning does not perform well when the dataset is small, this shortcoming is more apparent in dataset *BibSonomy*. The second reason is CDL only utilized item content to learn the features of items, while in our model we use not only capture the item but also the user with item content, user interests, and tags. (6) Our method EXPLORE outperforms all the baseline algorithms. The main reason is EXPLORE utilizes multi aspects of information. Compared to TRCF and CTR+, we can conclude that it is better to treat multi-aspects of content in a separate way rather than put them together.

4.4.2. Performance on tag recommendation

For tag recall, we set $M=$ 10, 20, 30, 40, and 50 respectively. Similar to item recall, from ~~the~~ part (b) of Figs. 6-8 we can find that: (1) PMF, SCF, and TAGCO achieve lower tag recall performances in all the three datasets. The main reason is that these three models only utilized rating information between items and tags. Meanwhile, the tag recalls of SCF and TAGCO increase slowly when M increases, especially in dataset *BibSonomy*. SCF is a k-neighborhood based algorithm and TAGCO is a co-occurrence based algorithm. They are count-based but there are limited numbers of neighbors and co-occurrences, which accounts for the poor performance of SCF and TAGCO. (2) CTR-SR outperforms PMF, SCF and TAGCO in most cases as CTR-SR utilizes item content. RBLT consistently outperforms CTR-SR in all three datasets, especially when M gets larger. The reason lies in that tags are utilized in topic models showing more semantic relevances in tag recommendation. (3) EXPLORE is significantly better than other methods in most cases. If we look further at the tagging density listed on Table 2 and the fluctuations of all the models' performances on three datasets, PMF, SCF, and TAGCO are more sensitive to the tag sparsity problem. They perform the worst in dataset *CiteUlike-t* that suffers the severest sparsity problem. CTR-SR and RBLT perform better than the former three models. EXPLORE performs stably on all three datasets, which is better than all the baseline models due to the utilization of multi-aspects of content and the integration with item recommendation.

4.4.3. Performance on co-recommendation metric

For rec-i tag recall, we also set M= 10, 20, 30, 40, and 50 respectively. For baseline algorithms, we take a Cartesian product of item recommendation algorithms and tag recommendation algorithms to form a set of new baselines to compare this metric. For example, *"TRCF & TAGCO"* means we use the results of TRCF and results of TAGCO to calculate $recall_{rec-i-t}$. As there are 30 combinations of item recommendation algorithms and tag recommendation algorithms, some baselines with poor performances are omitted in the figures due to the problem of clarity. From ~~the~~ part (c) of Figs. 6-8 we can see that: (1) our method EXPLORE consistently outperforms all the baselines in three datasets. EXPLORE performs better when M gets bigger, while some other baselines (e.g., *"CTR+ & TAGCO"* and *"TRCF & SCF"*) perform poorer when M increases. EXPLORE is more robust and stable. (2) Among the baselines, we can find *"CTR+ & RBLT"*, *"TRCF & RBLT"*, and *"TRCF & CTR-SR"* achieve better performances than other baselines, which shows the superiority of CTR-SR and RBLT. Similarly we ~~can~~ find SCF and TAGCO perform poorly in the $recall_{rec-i-t}$ metric. The reasons are similar to the analysis in Section 4.4.2: CTR-SR and RBLT utilize content information; SCF and TAGCO are count based algorithms, and the datasets have limited number of neighbors and co-occurrences. (3) Compared to $recall_{tag}$, we can see the differences between algorithms are more apparent in the co-recommendation metric $recall_{rec-i-t}$, indicating the $recall_{rec-i-t}$ is an effective metric to evaluate tag recommendation performance. Meanwhile, a higher $recall_{rec-i-t}$ indicates that the recommended tags provide more specific explanations to recommended items, and users will be more likely to accept these items.

From the above analysis we can see that EXPLORE shows superiority in both item recommendation and tag recommendation~~,~~ and outperforms baseline combinations in the co-recommendation metric.

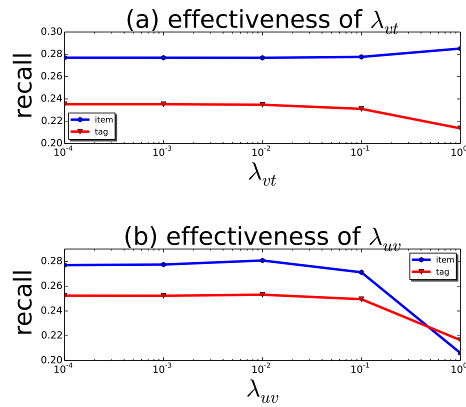
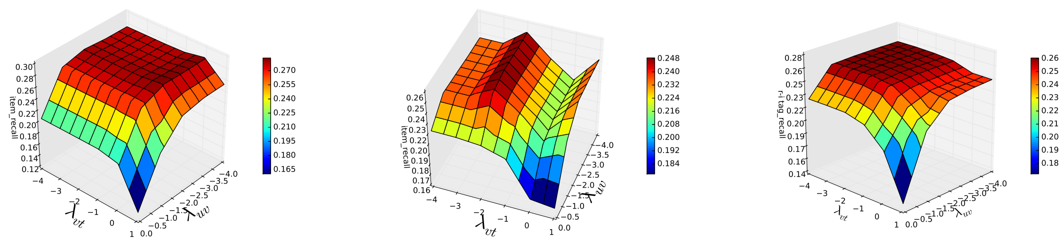


Figure 9: The marginal effects of λ_{uv} and λ_{vt} on *CiteULike-a*

4.5. Parameter Effect Analysis

To examine the sensitivity of EXPLORE to parameters λ_{uv} and λ_{vt}, we set $\lambda_u = 1.0$, $\lambda_v = 10.0$, $\lambda_t = 1.0$, $K = 200$, $a = 1$, $b = 0.01$, $a_t = 1$, and $b_t = 0.01$. We fix $\lambda_{uv} = 0$, $M = 50$ and vary λ_{vt} to see the item recall and tag recall, then we fix fix $\lambda_{vt} = 0$ and vary λ_{uv} to see the item recall and tag recall. Fig. 9 shows the results on *CiteULike-a*. From ~~the~~ Fig. 9 (a) we can see item recall and tag recall stay stable when λ_{vt} varies 0.001 to 0.01. As λ_{vt} increases from 0.01 to 1, the item recall increases while the tag recall decreases. ~~The~~ Fig. 9 (b) shows the item recall and tag recall first increase with λ_{uv} and then decrease after $\lambda_{uv} = 0.01$.

To further explore the joint effect of λ_{uv} and λ_{vt}, we conduct a grid search on these two parameters. The results on *CiteULike-a* are shown on Fig. 10. Fig. 10 (a) shows the $recall_{item}$@50 on *CiteULike-a*. We can see the item recall gets the best performance when $\lambda_{uv} = 0.01$ and $\lambda_{vt} = 1$, when λ_{uv} gets larger than 0.1 and λ_{vt} gets larger than 1, the item recall falls dramatically. Fig. 10 (b) shows the $recall_{tag}$@50 on *CiteULike-a*. We can see the tag recall gets the best performance when $\lambda_{uv} = 0.01$ and $\lambda_{vt} = 0.01$. The tag recall increases when λ_{vt} increases from 0.0001 to 0.01 and from 1 to 10, and it decreases when λ_{vt} increases from 0.01 to 1. The performance of tag recall stays stable when



(a) $recall_{item}@M$ (b) $recall_{tag}@M$ (c) $recall_{rec-i-t}@M$

Figure 10: Experimental results on *CiteULike-t* datasets. (a) shows the recall of item recommendation $recall_{item}@M$. (b) shows the recall of tag recommendation $recall_{tag}@M$. (c) shows the rec-item tag recall $recall_{rec-i-t}@M$

λ_{uv} stays between 0.0001 and 1 and falls dramatically after λ_{uv} increases over 0.1. Fig. 10 (c) shows the $recall_{rec-i-t}@50$ on *CiteULike-a*. The performance of $recall_{rec-i-t}@50$ stays stable when $\lambda_{uv} < 0.1$ and $\lambda_{vt} < 0.1$, it is a tradeoff between item recall and tag recall.

4.6. Interpretability and Explanability

Besides the superior recommendation performance, our proposed framework can also provide a very good interpretation. Two examples in Table 3 and Table 4 (see the appendix) show users with their top 3 topics along with their top 10 preferred articles predicted by EXPLORE and CTR respectively, the EXPLORE also provides the top 10 recommended tags for each corresponding articles. As shown in Table 3, User *I* is a biologist with a research interest on RNA, specifically RNA sequencing (RNA-Seq). This information is showed by the first two words in the first topic of EXPLORE and the ninth and tenth words of first topic of CTR. The precision of the top 10 articles for EXPLORE and CTR are 80 percent and 30 percent respectively for User *I*. Similarly, User *II* in Table 4 is interested in social networks. The precisions of EXPLORE and CTR are 90 percent and 40 percent respectively.

If we look further into the training set and the recommended tags, we can find more about user's interest and the correlation between his interest and the articles he rated. User *I* is a biologist interested in mRNA and isoform, he focuses on RNA sequencing and heads for the next generation sequencing. The tags *'isoform'*, *'mrna-seq'*, *'isoform'*, *'next-generation-sequenzing'*, *'altsplice'* proved this inference. From the recommendation result of EXPLORE, almost all ~~of~~ the top 10 articles are closely related to the specific interest of User *I*, while the recommended articles by CTR are from a larger domain related to *'gene activity'* and *'RNA-Seq'*, thus leading to a low precision. Similarly, User *II* is more likely to be a software designer focused on network learning, the top 10 articles recommended by EXPLORE contain social network theories and software design. From the attached recommended tags we can see the user focuses on *'personal-learning-ENV'*, *'c-sap'*, and *'e-learn'*. Meanwhile all the recommended articles of CTR are about social network except the one *'Social software: E-learning beyond learning management systems'*, which is about software design.

Moreover, the tags recommended by EXPLORE can provide explanations and help users in the decision making for the recommended articles. Think about the case when User *I* see the articles recommended to him by EXPLORE. He looks through the titles with tags and decide if he should read it in detail later. For example the article *'Mapping and quantifying mammalian transcriptomes by RNA-Seq'* with its recommended tags *'4countseq'*, *'mrna-seq'*, *'aziz'*, *'hesis-pmi'*, *'altsplice'*, *'doctoral-thesis'*, *'mapping-rnaseq'*, *'solid'*, *'isoform'*, and *'aligner'*. User *I* finds the title is in his research domain and the tags meet his specific interests, which will help him make the decision to read this article. While this case comes to CTR, User *I* only finds the title of the article, he doesn't know if this article meets his specific interest on *'mRNA-Seq'* rather than *'RNA-Seq'* as titles often failed to convey enough information. It will take some time for him to decide whether to read this article or not.

5. Conclusion and Future Work

In this paper, we have developed a novel hierarchical framework named EXPLORE that jointly recommends items and corresponding tags synchronously for explanations. We ~~have~~ explored multi-aspects of content information to capture the item more comprehensively with user's implicit feedback, item's contents, and item's tags. Comprehensive experimental results on the three real-world data sets show that our method EXPLORE outperforms both item recommendation methods and tag recommendation methods in prediction accuracy. EXPLORE also shows superiority in co-recommendation metrics against the combination of item and tag recommendation methods. We conclude that the item-tag co-recommendation in EXPLORE are of mutual promotion, and EXPLORE can better alleviate the cold start problem of recommender systems and provide explanations of recommendation results to users.

In the future, we intend to extend our work in two main directions: one is to design new methods to balance the performance of item recommendation and tag recommendation. The other direction is to add more constraints and auxiliary information into consideration, such as time, user-item subgroups, and social connections.

References

Table 3: Interpretability of ~~the~~ EXPLORE for user I

EXPLORE	User I	in lib?
top 3 topics	1. rna, sequence, gene, expression, humans, dna, analysis, mice, animals, genetic	
	2. humans, dna, chromatin, cells, animals, proteins, genetic, cell, mice, gene	
	3. health, humans, medical, internet, information, care, education, clinical, medicine, systems	
top 10 articles	1. RNA-Seq: a revolutionary tool for transcriptomics.	yes
	2. Mapping and quantifying mammalian transcriptomes by RNA-Seq	yes
	3. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation.	no
	4. TopHat: discovering splice junctions with RNA-Seq	yes
	5. RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays.	yes
	6. Statistical inferences for isoform expression in RNA-Seq	yes
	7. Stem cell transcriptome profiling via massive-scale mRNA sequencing	yes
	8. Computation for ChIP-seq and RNA-seq studies	yes
	9. Alternative isoform regulation in human tissue transcriptomes.	yes
	10. ChIP-seq: advantages and challenges of a maturing technology.	no
top 10 tags for each articles	1. mrna-seq, thesis-pmi, aziz, next-generation-sequenzing, isoforms, 4countseq, technology-figures ngs-technical, metais-embrapa, sequencing-technologie	
	2. 4countseq, mrna-seq, aziz, thesis-pmi, altsplice, doctoral-thesis, mapping-rnaseq, solid, isoform, aligner	
	3. thesis-pmi, altsplice, isoform, 4countseq, isoforms, aziz, mrna-seq, mapping-rnaseq, rna-seq-analysis, solid	
	4. 4countseq, altsplice, isoform, mapping-rnaseq, thesis-pmi, aziz, isoforms, solid, mrna-seq, aligners	
	5. eda-template, rma, marcel, tech-reprod, rna-seq-analysis, microarray-technology, microarray-normalization agilent, seq-processing, maqc	
	6. isoform, altsplice, isoforms, thesis-pmi, aziz, 4countseq, mrna-seq, rna-seq-statistics, mapping-rnaseq, rna-seq-analysis	
	7. 4countseq, stem-cell, deep-sequencing, esc, mrna-seq, short, aligner, mapping-rnaseq, aziz, new-technology	
	8. seq-processing, next-generation-sequenzing, nextgen-sequencing, technology-figures, dna-sequencing, metais-embrapa dna-sequencing, ngs-technical, analysis-tool, nextgen-seq	
	9. altsplice, isoform, thesis-pmi, isoforms, 4countseq, mrna-seq, aziz, mapping-rnaseq, as, alternative-splicing	
	10. next-generation-sequenzing, ngs-technical, chromatin-structure, nextgen-sequencing, seq-processing technology-figures, ngs-review, solexa, replicates, peak-calling	
CTR	**User I**	**in lib?**
top 3 topics	1. transcripts, splicing, alternative, genes, transcriptome, exons, rnaseq, transcript, rna, sequencing	
	2. sequencing, reads, sequence, genome, assembly, dna, short, technologies, read, nextgeneration	
	3. dna, methylation, modifications, modification, epigenetic, histone, chromatin, etal, cpg, hkme	
top 10 articles	1. Most "Dark Matter" Transcripts Are Associated With Known Genes	no
	2. Mapping and quantifying mammalian transcriptomes by RNA-Seq	yes
	3. A global view of gene activity and alternative splicing by deep sequencing of the human transcriptome.	no
	4. Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing	no
	5. RNA-Seq: a revolutionary tool for transcriptomics.	yes
	6. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation.	no
	7. Dynamic repertoire of a eukaryotic transcriptome surveyed at single-nucleotide resolution	no
	8. Ab initio reconstruction of cell typeCspecific transcriptomes in mouse reveals the conserved multi-exonic structure of lincRNAs	no
	9. TopHat: discovering splice junctions with RNA-Seq	yes
	10. Novel RNAs identified from an in-depth analysis of the transcriptome of human chromosomes 21 and 22.	no

Table 4: Interpretability of the EXPLORE for user II

EXPLORE	User II	in lib?
top 3 topics	1. phylogeny, evolution, sequence, likelihood, models, biological, dna, functions, analysis, alignment	
	2. sequence, software, analysis, databases, biology, computational, genetic, gene, expression, genomics	
	3. web, semantic, knowledge, ontology, services, rdf, information, management, ontologies, representation	
top 10 articles	1. Wikis, blogs and podcasts: a new generation of Web-based tools for virtual collaborative clinical practice and education	no
	2. From VLEs to learning webs: the implications of Web 2.0 for learning and teaching	yes
	3. Social software: E-learning beyond learning management systems	yes
	4. Social network sites: Definition, history, and scholarship	yes
	5. The Benefits of Facebook "Friends:" Social Capital and College Students' Use of Online Social Network Sites	yes
	6. Personal Learning Environments - the future of eLearning?	yes
	7. The 'digital natives' debate: A critical review of the evidence	yes
	8. Friends, friendsters, and top 8: Writing community into being on social network sites	yes
	9. Learning Networks and Connective Knowledge	yes
	10. Designs for network learning: a communities of practice perspective	yes
top 10 tags for each articles	1. personal-learning-ENV, ple, podcasts, c-sap, medical-edu, e-learn, ot-web, podcast, blended-learning, educacion	
	2. personal-learning-ENV, ple, c-sap, e-learn, blended-learning, higher-edu, distance, informal-learning, podcasts, podcast	
	3. personal-learning-ENV, ple, c-sap, e-learn, blended-learning, informal-learning, higher-edu, distance, teachers, podcast	
	4. onlinesocialnetworks, lukehutton, opw, online-social-networks, socialnetworkingsites, tam, online-research-methods, network-dynamics, acceptance, tie-strength	
	5. onlinesocialnetworks, lukehutton, self-reporting, tie-strength, myspace, weak-ties, tie, online-social-networks, homophily, social-networking-site	
	6. personal-learning-ENV, ple, c-sap, e-learn, blended-learning, distance, instructional-design, podcasts, informal-learning, higher-education	
	7. digital-divide, digital-divide, skills, higher-edu, uolpedr, tam, recommended, online-edu, educational, information-society	
	8. onlinesocialnetworks, lukehutton, self-reporting, myspace, tie-strength, tie, online-social-networks, weak-ties, homophily, facebook	
	9. personal-learning-ENV, ple, c-sap, e-learn, blended-learn, distance, instructional-design, informal-learning, podcasts, weblogs	
	10. personal-learning-ENV, ple, c-sap, e-learning, blended-learning, distance, weblogs, informal-learning, instructional-design, podcasts	
CTR	**User II**	**in lib?**
top 3 topics	1. learning, students, education, educational, teaching, teachers, student, literacy, online, skills	
	2. community, communities, online, social, communication, blogs, members, sites, users, groups	
	3. copyright, abstract, wiley, print, users, original, author, applies, individual, writing	
top 10 articles	1. Is online instruction perceived as effective as campus instruction by graduate students in education?	no
	2. The Influence of Teachers' Technology Use on Instructional Practices	no
	3. Social software: E-learning beyond learning management systems	yes
	4. Social network sites: Definition, history, and scholarship	yes
	5. The Benefits of Facebook "Friends:" Social Capital and College Students' Use of Online Social Network Sites	yes
	6. The 'digital natives' debate: A critical review of the evidence	yes
	7. Learning and Teaching Styles in Engineering Education	no
	8. Learning Styles: Concepts and Evidence.	no
	9. The future of eLearning.	no
	10. The impact of web-logs (blogs) on student perceptions of isolation and alienation in a web-based distance-learning environment	no