Tackling Data Sparseness in Recommendation Using Social Media Based Topic Hierarchy Modeling

Xingwei Zhu[1], **Zhao-Yan Ming**[2*], **Yu Hao**[1] **and Xiaoyan Zhu**[1]
[1]State Key Laboratory of Intelligent Technology and Systems, Tsinghua National Laboratory for Information Science and Technology, Department of Computer Sci. and Tech., Tsinghua University
[2]Department of Computer Science, DigiPen Institute of Technology
etzhu192@hotmail.com, mingzhaoyan@gmail.com, haoyu@mail.tsinghua.edu.cn, zxy-dcs@tsinghua.edu.cn

Abstract

Recommendation systems play an important role in E-Commerce. However, their potential usefulness in real world applications is greatly limited by the availability of historical rating records from the customers. This paper presents a novel method to tackle the problem of data sparseness in user ratings with rich and timely domain information from social media. We first extract multiple side information for products from their relevant social media contents. Next, we convert the information into weighted topic-item ratings and inject them into an extended latent factor based recommendation model in an optimized approach. Our evaluation on two real world datasets demonstrates the superiority of our method over state-of-the-art methods.

1 Introduction

The recommendation systems in e-commerce sites such as eBay [1] and Amazon [2] play a key role in understanding the user purchasing behaviors. Among existing recommenders, the latent factor based collaborative filtering model [Koren and Bell, 2011] that makes use of known user-product ratings to predict the unknown ratings has been shown to be effective. However, the problem of sparse user ratings limits its potential usefulness [Popescul and Ungar, 2001][Zhang *et al.*, 2013]. Side information that helps to establish more elaborated relation between users and products, such as item tags [Tso-Sutter *et al.*, 2008] and linked data [Ostuni *et al.*, 2013] have been shown to be useful in supplementing the sparse rating data. However, the above side information can be hard to obtain as well. Moreover, their coverage on the types of products may not be comprehensive either.

Social media contents that are rapidly growing in the recent years give us another angle of solving the data sparseness problem. Given the huge user bases and the active user participation, social media sites like Twitter [3] and Facebook [4] provide abundant user contributed contents that potentially contain information about many types of products.

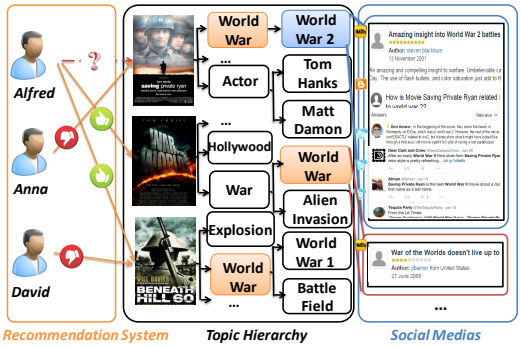

Figure 1: Topic hierarchies could help to understand a user (e.g., Alfred)'s rating behavior on movies using the fine-grained user generated topics (e.g., "World War", "World War 2"), topic relations on the hierarchy and frequency of topics mentioned in social media.

In this paper, we propose a novel framework that uses a product's related social media contents to establish a deeper understanding of it. Specifically, we propose the *Topic Hierarchy based Recommendation(THRec) model*, in which we model social media contents using topic hierarchies and inject the derived topic-item information to enrich the links between users and items. Compared to the raw social media contents as used in [McAuley and Leskovec, 2013], a topic hierarchy [Zhu *et al.*, 2013] can provide a fine-grained topic-level view of a social media corpus, in which the domain topics and topic relations will complement the sparse user rating data. For example, As shown in Figure 1, when recommending movies to the user Alfred, the shared topics, e.g., "World War" among the topic hierarchies of his rated movies could reveal his personal favor on movies. Moreover, given the tight relatedness between "World War 2" and "World War" on the hierarchy and "World War 2"'s high popularity in social media, we can also recommend Alfred with movies about "World War 2" such as "Saving Private Ryan".

However, to incorporate the information from products' topic hierarchies into a recommendation model is not trivial. We identify two key challenges in this research.

- Compared to user-item ratings, the information within items' topic hierarchies like topic terms has very dif-

* Corresponding author.
[1]http://eBay.com
[2]http://www.amazon.com
[3]http://twitter.com
[4]http://www.facebook.com

ferent data nature, making it difficult to combine them together to train a unified recommendation model.

- Given the large volume and high noise of social media contents, the items' topic hierarchies may contain a large number of topics with uneven quality. How to use this information effectively is also challenging.

To address the above challenges, we propose a novel latent factor based model, where we complement the user-item ratings with topic hierarchy derived topic-item ratings that come in abundance from the social media data. Next, in order to optimize the impact of the topic-item ratings on the recommendation model, we evaluate the importance of each topic's ratings using a topic weight and propose to learn the weights from the ratings directly. To this end, we design a user-topic consistency factor to adjust topic weights to best fit the real users' rating behaviors. A topic-topic consistency factor is also employed to refine the topic weights based on topics' semantic relatedness. To summarize, the main contributions of this research are as follows:

- We propose an approach of using social media contents to tackle data sparseness in recommendation systems, which explores the usefulness of multiple topic hierarchy derived side information.
- We design a novel latent factor based recommendation model, in which the weights of different side information can be learned to optimize their impacts on the recommendation results.

2 Related Work

In modern E-Commerce, recommender systems play key roles in helping users to find their potentially desired items. In previous research works, various kinds of recommenders were introduced for this task, including content-based [Moshfeghi *et al.*, 2011], knowledge-based [Ziegler *et al.*, 2004] and collaborative filtering (CF) [Su and Khoshgoftaar, 2009] recommenders. Among them, the latent factor based CF recommender [Koren and Bell, 2011] has received much attention in both the academic and industrial communities due to its high performance and ease of use.

However, the performance of latent factor based recommender is heavily affected by the sparseness of dataset [Popescul and Ungar, 2001]. To address this problem, techniques like co-clustering [Leung *et al.*, 2011] and community detection [Zhang *et al.*, 2013] have been introduced in previous works. On the other hand, some researchers also proposed to solve this problem by enhancing recommenders with external side information like user reviews [McAuley and Leskovec, 2013], item tags [Rafailidis *et al.*, 2014] and knowledge bases [Ostuni *et al.*, 2013]. However, such high quality external data is usually hard to obtain. Moreover, many of these approaches assumed that the external information is of high quality, which may no longer stand when the information is harvested from the noisy internet. Although in research efforts like collective matrix factorization [Singh and Gordon, 2008][Park *et al.*, 2013], manually-defined weights were introduced to measure the data quality, their effectiveness relied on the specific heuristic rules behind the weight estimation and could vary a lot on different datasets.

In this paper, we propose to use social media based topic hierarchies to enhance recommenders. Generally, topic hierarchy generation methods has been studied in many previous works [Wang *et al.*, 2014][Zhu *et al.*, 2013][Ming *et al.*, 2010]. However, according to our limited knowledge, the application of this technique is few.

3 Item Modeling using Topic Hierarchies

In this section, taking advantage of the rich and timely information within social media, we collect topic hierarchy derived side information for items from their relevant social media contents to enhance user ratings for recommendation.

3.1 Topic Hierarchy Model

Denoting i as an item in the recommendation system and $\mathcal{D}_i$ as the corpus of its relevant social media contents, its topic hierarchy $\mathcal{H}_i$ is a tree rooted at i and consists of the following two components:

- Topic set $\mathcal{T}_i$: each $t_k \in \mathcal{T}_i$ is a noun phrase, indicating a subtopic of the item i. Specifically, the item's name $i \in \mathcal{T}_i$ is the unique root topic and each non-root topic must have one parent topic in $\mathcal{T}_i$.
- Content set $\mathcal{C}_i$: for each $c_{t_k} \in \mathcal{C}_i$, $c_{t_k} \subset \mathcal{D}_i$, is the set of relevant social media contents of the corresponding subtopic $t_k \in \mathcal{T}_i$.

Generally, the topic set $\mathcal{T}_i$ can provide a compact, fine-grained hierarchical view of the user generated subtopics for item i. For example, given the movie "Saving Private Ryan", its topic set may contain topics such as "Tom Hanks", "World War" and "World War 2", where "World War 2" is a subtopic of "World War". Besides, the content set $\mathcal{C}_i$ also offers useful description for each topic. Particularly, the size of c_{t_k} may reflex the popularity of topic t_k in social media.

3.2 Topic Hierarchy Construction for Items

In this paper, we employ the method described in [Zhu *et al.*, 2013] for topic hierarchy construction. Specifically, regarding each item i as a root topic, we collect its related blogs, reviews and tweets as $\mathcal{D}_i$. Next, the item's subtopics and topic relations are extracted from $\mathcal{D}_i$. Then a graph based method is used to construct item i's topic hierarchy incrementally using the extracted topics. Due to the limited space, please refer to [Zhu *et al.*, 2013] for more details.

Finally, denoting $\mathcal{I} = \{i_1, i_2, ...\}$ as the set of all items, a topic hierarchy set $\mathcal{H}(\mathcal{I}) = \{\mathcal{H}_{i_1}, \mathcal{H}_{i_2}, ...\}$ can be obtained and updated off-line using the presented process.

3.3 Topic Hierarchy based Side Information

The topic hierarchies in $\mathcal{H}(\mathcal{I})$ bear rich information to harvest for recommendation. Firstly, inspired by [Tso-Sutter *et al.*, 2008], we can regard item i's subtopics on the hierarchy as its features directly. Besides, in this paper, we further investigate the following two kinds of side information within topic hierarchies, i.e., topic relatedness and topic popularity.

Topic Relatedness

The semantic relatedness between topics can identify the relations of their associated items, hence improving the recommendation results on them. In the topic hierarchy set, the connectivity of two topics reflexes the strength of their relatedness. For instance, in the movie domain, since "Pixar" is a subtopic of "Animation Studio" on many topic hierarchies, their semantic relatedness should be strong. On the contrary, the relatedness of "Pixar" and "Graphic file" [5] is weak since there is few connections between them for movies.

In this research, we first estimate $\#sub(t_m, t_n) = \sum_{i \in \mathcal{I}} sub_i(t_m, t_n)$, in which $sub_i(t_m, t_n) = 1$ *iff* topic t_n is a subtopic of t_m on $\mathcal{H}_i$; otherwise $sub_i(t_m, t_n) = 0$. Next, the semantic relatedness from t_m to its subtopic t_n, i.e., $s(t_m, t_n)$ is calculated as follows:

$$s(t_m, t_n) = \begin{cases} \frac{\#sub(t_m, t_n)}{\sum_k \#sub(t_m, t_k)} & , \text{ if } \#sub(t_n, t_m) = 0 \\ 0 & , \text{ otherwise.} \end{cases} \tag{1}$$

Finally, for each $s(t_m, t_n) \neq 0$, we let $s(t_n, t_m) = s(t_m, t_n)$ to guarantee the symmetry of topic relatedness.

Topic Popularity

The popularity of topics reflexes the ever-changing common interests of all users, which may also help in recommendation. Due to the timeliness of social media contents, it is particularly convenient to capture the topic popularity from topic hierarchies. For example, a rapid growth of the content set for topic "Leonardo DiCaprio" can indicate that this subtopic is becoming popular recently. In this research, we estimate the popularity of t_m, i.e., $p(t_m)$ as follows:

$$p(t_m) = \frac{\#doc(t_m)}{\sum_{k \in \mathcal{T}(\mathcal{I})} \#doc(t_k)} \tag{2}$$

where $\mathcal{T}(\mathcal{I}) = \bigcup_{i \in \mathcal{I}} (\mathcal{T}_i - \{i\})$ [6] and $doc(t_m) = \bigcup_{i \in \mathcal{I}} c_{i,t_m}$, in which c_{i,t_m} denotes topic t_m's content sets on $\mathcal{H}_i$ and $\#doc(t_m)$ indicates the number of documents (e.g., tweets) in $doc(t_m)$.

4 Topic Hierarchy based Recommendation

In this section, we first give an introduction to the latent factor based recommendation model. Next, we extend it to our proposed THRec model with the extracted side information from topic hierarchies.

4.1 Latent Factor based Recommendation Model

Recall that $\mathcal{I}$ is the item set for recommendation and denote $\mathcal{U}$ as the set of the target users. The "standard" latent factor based recommendation model [Koren and Bell, 2011] predicts the rating $r_{u,i}$ on item $i \in \mathcal{I}$ given by user $u \in \mathcal{U}$ using the following formula,

$$r_{u,i} = \mu + b_u + b_i + p_u^T q_i \tag{3}$$

[5] "Pixar" is also a graphic file format in the Computer Graphics domain.

[6] The reason to exclude the root topic i is that since i is only contained by the corresponding $\mathcal{H}_i$, it is too specific to indicate the items' properties or users' interests.

in which μ indicates the overall rating offset. b_u and b_i are the user and item rating biases. q_i and p_u are two vectors that represent the latent factors of item i and user u, respectively. Given a rating corpus $\mathcal{R}$, all these parameters can be estimated by minimizing the following objective function,

$$\Phi = \sum_{r'_{u,i} \in \mathcal{R}} (r'_{u,i} - r_{u,i})^2 + \lambda \Omega_{u,i} \tag{4}$$

in which $r'_{u,i}$ indicates a rating sample in $\mathcal{R}$ and $\Omega_{u,i} = ||q_i||_2^2 + ||p_u||_2^2 + ||b_u||_2^2 + ||b_i||_2^2$, is the regularizer. With the estimated parameters, we can predict a user's rating on any item using Equation 3 straight-forwardly. However, a fine estimation of so many parameters requires a large amount of training data, which is usually not available due to the sparseness of the historical rating records of real users.

4.2 Topic Hierarchy based Recommendation Model

In this section, we enhance the latent factor based recommender with the side information from topic hierarchies. However, it is difficult to combine the user ratings with items' subtopics directly due to their different data nature. To solve this problem, we adopt a new perspective in which a topic is regarded as a pseudo user. Then for each topic, an item's topic hierarchy is converted into a set of special ratings, i.e., topic-item ratings, which are high only if the hierarchy includes the corresponding topic. As a result, the topic-item ratings of a topic can capture a specific kind of personal taste of real users. For example, since the topic "War film" only gives high ratings to movies that contain this topic, it will reflex the rating behavior of a "perfect" war movie lover. Formally, given topic $t \in \mathcal{T}(\mathcal{I})$ and item $i \in \mathcal{I}$, we define a topic-item rating $y'_{t,i}$ as:

$$y'_{t,i} = \begin{cases} 1 & , \text{ if } t \in \mathcal{T}_i \\ 0 & , \text{ otherwise.} \end{cases} \tag{5}$$

Next, similar to the user-item ratings in Equation 3, the following $y_{t,i}$ is used to predict a topic t's rating on an item i,

$$y_{t,i} = \mu + b_t + b_i + p_t^T q_i \tag{6}$$

in which b_t and p_t are the pseudo user t's rating bias and latent factor representation, respectively. It is worthy noting that the parameters μ, b_i and q_i are shared by both Equation 3 and 6. Therefore the information within topic-item ratings can help to characterize real users through their shared items. More importantly, since all the topic-item ratings can be explicitly determined by Equation 5, different from user-item ratings, they won't suffer from the problem of data sparseness.

However, due to the uneven quality of contents in social media, the topic set $\mathcal{T}(\mathcal{I})$ may contain useless or even misleading topics (e.g., "Awesome movie" for movies), which could potentially damage the recommendation performance. To tackle this problem, previous work mainly relies on heuristic rules to determine the weights of different side information [Singh and Gordon, 2008][Cheng *et al.*, 2014]. In this research, we propose a novel approach, i.e., the Topic Hierarchy based Recommendation(THRec) model, which can learn

the weights from user ratings directly. Specifically, denoting a topic weight vector $w = [w_{t_0}, w_{t_1}, ..., w_{t_{|\mathcal{T}(\mathcal{I})|}}]$, in which w_{t_k} indicates the impact of topic t_k's topic-item ratings on the recommendation model, we employ the following two assumptions for the weight estimation:

Topic-User Consistency: A topic's weight should be consistent with its influence on real users. For example, if the topic-item ratings of a topic, e.g., "World War", are very similar to those of real users, e.g., war movie fans, assigning it with high weight will help to capture these users' common interests, hence improving the recommendation results for them.

Topic-Topic Consistency: A topic's weight should be consistent with those of its related topics. For example, if we know that "Disney" and "Pixar" are relevant to each other in many aspects (e.g., they are both animation studios) in the movie domain, their weights, i.e., impacts on the recommendation results, should be also similar.

Based on the assumptions, in the proposed THRec model we extend the original objective function of latent factor based recommendation model in Equation 4 as follows:

$$\min(\underbrace{\sum_{r'_{u,i} \in \mathcal{R}} (r'_{u,i} - r_{u,i})^2 + \sum_{y'_{t,i} \in \mathcal{R}_t} w_t \cdot (y'_{t,i} - y_{t,i})^2}_{\text{Topic-User Consistency Factor}} + \underbrace{\sum_{t_m, t_n \in \mathcal{T}(\mathcal{I})} s(t_m, t_n) \cdot (w_{t_m} - w_{t_n})^2}_{\text{Topic-Topic Consistency Factor}} + \underbrace{\lambda(\Omega_{u,y,i} + ||w||_2^2)}_{\text{Regularizer}}) \quad (7)$$

$$\text{s.t.} \quad \sum_{t \in \mathcal{T}(\mathcal{I})} w_t = L, 0 \leq w_t \leq 1$$

in which $\mathcal{R}_t$ indicates the set of topic-item ratings and $s(t_m, t_n)$ is the estimated topic relatedness between t_m and t_n. $\Omega_{u,y,i} = ||q_i||_2^2 + ||b_i||_2^2 + ||p_u||_2^2 + ||b_u||_2^2 + ||p_t||_2^2 + ||b_t||_2^2$, is the regularizer and the parameter L in the constraint is an adjustable hyper-parameter. Generally, the extended objective function contains two major parts which capture the topic-user and topic-topic consistency, respectively. The following remarks will explain how the THRec model optimizes the recommendation performance using the topic weights:

- When the topic weights are known, the topic-user consistency factor becomes the same as in collective matrix factorization model [Singh and Gordon, 2008]. However, when the latent factors are known, the THRec model can also learn the topic weights by minimizing the difference of rating behaviors between topics and users.
- In the topic-topic consistency factor, the topics' semantic relatedness $s(t_m, t_n)$ is employed to identify highly related topics. Generally, this factor refines the topic weights by minimizing the difference of weights between topics that have high topic relatedness.
- In the constraint, all topic weights are limited to $[0, 1]$, which will never excess that of a real user. This is reasonable since our goal is to recommend items to only real users. The hyper-parameter L indicates the overall importance of all topics. In practice, it can be determined by either tuning or using prior knowledge. Generally, the higher L is, the topic-item ratings will have more affect on the system.

4.3 Parameter Estimation

We adopt the stochastic gradient descent (SGD) to solve the objective function in Equation 7. Specifically, when learning the latent factors of users, items and topics, i.e., μ, b_u, b_t, b_i, p_u, p_t and q_i, we fix the topic weights and update the latent factors as in traditional collective matrix factorization model using both user-item and topic-item ratings.

To learn the topic weights, we first initiate the weight of each topic based on its popularity as follows:

$$w_{t_m} = p(t_m) \cdot L \quad (8)$$

The intuition behind is that, popular topics in a specific domain are usually also important topics. Next, for each topic-item rating $y'_{t,i} \in \mathcal{R}_t$, we fix all the latent factors and update the corresponding topic weight w_t as follows:

$$w_t \leftarrow w_t - \gamma(\lambda \quad w_t \quad + (y'_{t,i} - y_{t,i})^2 + 2 \quad \cdot \sum_{t' \in \mathcal{T}(\mathcal{I})} s(t, t')(w_t - w_{t'})) \quad (9)$$

in which γ is the learning rate. Note that the updated w will no longer satisfy the weight constraints. To solve this problem, we first adopt Equation 10 to guarantee that the sum of the weights still equals L. Next, if any of the resultant topic weight is out of $[0, 1]$, Equation 11 is used to fix this outlier. We rerun this process until both constraints are satisfied.

$$w_u = \frac{L w_u}{\sum_{u \in \mathcal{T}(\mathcal{I})} w_u} \quad (10)$$

$$w_u = \begin{cases} 0 & : \quad w_u < 0 \\ 1 & : \quad w_u > 1 \\ w_u & : \quad \text{otherwise.} \end{cases} \quad (11)$$

5 Evaluation

5.1 Experimental Setup

We evaluate the performance of the THRec model on two datasets. The first is the MovieLens 1M dataset ($Movie$) [7]. It contains one million ratings on $3,706$ movies produced by $6,040$ users. The second dataset we used is an iTunes app rating dataset (App). It contains $88,253$ ratings on $1,485$ apps produced by $4,483$ users.

For the items in both datasets, we collected their relevant blogs, reviews and tweets to form the social media corpus. To generate topic hierarchies from the crawled social media contents, the method described in [Zhu *et al.*, 2013] was used. A brief statistic on the social media corpus and the generated topic hierarchies for items is presented in Table 1.

For the evaluation, we first split the user-item ratings into two parts, i.e., 80% for model training and 20% for test. Next, we generate training datasets of different data sparseness by randomly removing ratings from the full training set.

[7] grouplens.org/datasets/movielens/

Data Set	*Social Media Corpus*			*Topic Hierarchy Set*	
	blogs	review	tweet	#topic	#edges
App	88,045	2,307,317	2,089,968	5,610	9,214
Movie	236,208	1,202,259	1,947,267	2,044	1,401

Table 1: Statistics on the collected social media corpus for items and the generated topic hierarchy sets.

Specifically, we limit at most k ratings for each user, where $k = 5, 10, 20, 30$ and ∞ (i.e., the full training set), resulting in five training datasets. Generally, the dataset is more sparse when its corresponding k is smaller. Finally, we adopt MyMediaLite [8] toolkit to implement both our THRec model and the baseline methods [9]. *Root Mean Square Error* ($RMSE$) is adopted to measure the recommendation performance.

5.2 Impact of Topic-Item Ratings

In this section, we evaluate the impact of the topic hierarchy derived topic-item ratings on the proposed model. Recall that the parameter L controls the overall impacts of the topic-item ratings, we first investigate the performance of the THRec model with different L on all datasets. The experimental results are shown in Figure 2.

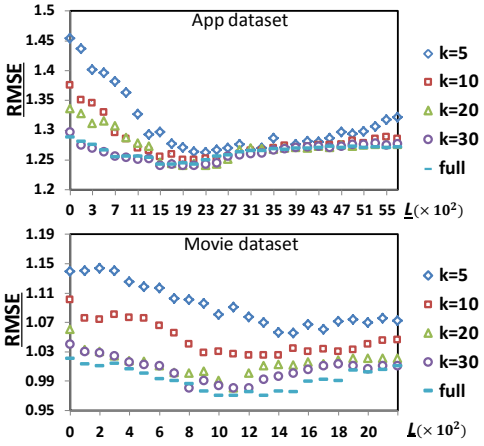

Figure 2: The performance of THRec model for different L on the two datasets of different data sparseness.

From the results we have the following two major conclusions: first, compared to the performance when $L = 0$, thus no information from the topic hierarchy is used, the proposed model performs significantly better when the topic-item ratings are utilized on both datasets for all k. Specifically, the largest $RMSE$ improvements, i.e., 13.6% and 7.2% are achieved when the training data is the sparsest, i.e., $k = 5$ on App and Movie dataset, respectively. This result demonstrates the usefulness of the side information within topic hierarchies on tackling the problem of data sparseness. Second, we can see that larger L does not always lead to better recommendation results. This is reasonable since the negative impacts of those misleading topics in the social media could also be amplified when L is large. From this observation, we can see the necessity of the topic weight learning algorithm in distinguishing the useful topics from those of low quality.

Table 2 lists some example topics on the movie domain ranked by the learned topic weights with the optimized L, which demonstrates the effectiveness of the proposed topic weight learning method. Firstly, we can see that the top ranked topics are all critical for movie recommendation. For example, the topic-item ratings of "War film" can help to recommend movies to many war movie lovers. On the other hand, the learned low topic weights can also limit the potentially negative impact of the low quality topics such as "Awesome movie", which could mislead a sci-fi movie lover to watch an awesome history movie.

	Topics on movie domain
Top 10	Steven Spielberg, Akira, George Lucas, Steven, Technique, Special effects, Oscar, James Bond, Action movies, War film
Bottom 10	Awesome movie, Very addictive, Cool, Best movie, Stinks, Dumb, Time waster,Alright, Haha, Complaint

Table 2: The top and bottom 10 topics on the full movie dataset ranked by the learned topic weights.

5.3 Impact of Topic Relatedness and Popularity

In this section, we evaluate the usefulness of the two novel side information introduced in this paper, i.e., the topic relatedness and popularity. To this end, we compare the performance of our full model (THRec) with (1)THRec-tp, in which the topic popularity is discarded by initiating all topic weights uniformly, (2)THRec-tr, in which the topic relatedness is discarded by removing the topic-topic consistency factor in Equation 7 and (3) THRec-trp, in which both topic relatedness and popularity are discarded.

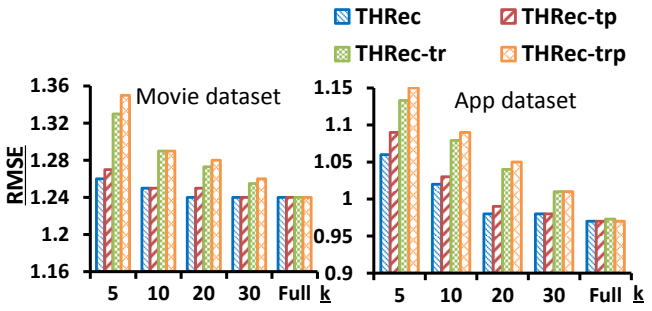

Figure 3: The performance of THRec model with different combinations of topic hierarchy derived side information.

Figure 3 shows the experimental results. We can see that our full model outperforms the THRec-tr and THRec-trp methods the most by 5.8% and 7.3%, respectively when $k = 5$. This observation demonstrates the usefulness of the topic relatedness in combating the data sparseness. However, the $RMSE$ improvement of the full model against the

[8] http://mymedialite.net/documentation/index.html

[9] Our datasets and codes are available at data.csaixyz.org/ijcai-2015/ijcai data.rar

THRec-tp method is not significant. By looking into the data, we find that it is mainly caused by the limited size of our collected social media content set, which leads to poor initiation of topic weights for some important but narrow topics like "War film" on the movie domain.

5.4 Comparison with State-of-the-art Methods

In this section, we compare the full THRec model with the following baseline methods:

- Collective Matrix Factorization [Singh and Gordon, 2008]. In this paper, we apply it to extend the well-known PMF [Mnih and Salakhutdinov, 2007] and SVD++ methods [Koren, 2008] with topic-item ratings, resulting in ex-PMF and ex-SVD, respectively.
- Tag-extended Recommender (TagRec) [Tso-Sutter *et al.*, 2008], in which item tags and user-item rating are combined by a neighborhood based recommender. In this paper, we used the topics extracted from the topic hierarchies as item tags.
- LDA-Matrix Factorization (LDAMF) [McAuley and Leskovec, 2013], in which a LDA model generated from item reviews is utilized to enhance the recommendation model. For the sake of fair comparison, we used all our collected social media contents for its LDA training.

The evaluation results are shown in Table 3 (App) and Table 4 (Movie). We can see that the proposed method outperforms all the baseline methods on both datasets, especially when $k = 5$, i.e., the user-item rating data is the sparsest. Compared to the ex-PMF and ex-SVD methods, taking advantage of the learned topic weights, the proposed method can make better use of the noisy social media contents, resulting in 19.6% and 13.8% average improvements over the two methods, respectively. Besides, the average 9.0% and 9.1% $RMSE$ gains of our method over the TagRec method suggest the superiority of the latent factor based recommenders. Finally, the proposed method also outperforms the LDAMF method significantly. The largest improvements, i.e., 7.4% on App and 7.8% on Movie dataset are observed when $k = 5$, for which the major reasons could be two fold. First, compared to very coarse topics generated by a LDA model, fine-grained topics extracted from topic hierarchies can better capture the various aspects of items; Second, the THRec method can also make better use of the relations between topics.

Method	k = 5	k = 10	k = 20	k = 30	full
ex-PMF	1.79	1.71	1.63	1.62	1.59
ex-SVD	1.63	1.52	1.46	1.44	1.44
TagRec	1.41	1.38	1.36	1.35	1.35
LDAMF	1.36	1.33	1.32	1.29	1.28
THRec	**1.26**†	**1.25**†	**1.24**†	**1.24**†	**1.24**†

Table 3: Comparison between our method and state-of-the-art methods on App dataset. † indicates significant improvement over all baseline methods (t-test, p-value < 0.01).

5.5 Comparison with other Side Information

Topic hierarchies play an important role in the THRec model. In this section, we investigate its advantage by comparing it with two commonly used side information, i.e., item tags and linked data. However, since we can find few such data for Apps, the following evaluation is only conducted on the movie dataset.

Method	k = 5	k = 10	k = 20	k = 30	full
ex-PMF	1.35	1.22	1.12	1.08	1.05
ex-SVD	1.28	1.17	1.08	1.05	1.03
TagRec	1.21	1.15	1.09	1.05	1.01
LDAMF	1.15	1.07	1.01	0.99	0.98
THRec	**1.06**†	**1.02**†	**0.98**†	**0.98**†	**0.97**†

Table 4: Comparison between our method and state-of-the-art methods on Movie dataset. † indicates significant improvement over all baseline methods (t-test, p-value < 0.05).

For each movie, we collect the genre tags in their MovieLens metadata as their tag sets. As to the linked data, we adopt the method introduced in [Ostuni *et al.*, 2013] to obtain the movies' relevant entities and relations from DBpedia [10]. Next, in order to inject these side information into the THRec framework, we regard each tag/entity as a pseudo user and convert them into $6,408/10,951$ tag-movie/entity-movie ratings. Finally, we replace the topic-movie ratings used in our model with these ratings, resulting in two baseline methods, i.e., THRec-tag, THRec-link, respectively. Table 5 illustrates their $RMSE$ performances on the movie dataset.

Method	k = 5	k = 10	k = 20	k = 30	full
THRec-tag	1.20	1.11	1.07	1.03	1.03
THRec-link	1.14	1.09	1.02	1.00	**0.97**
THRec	**1.06**†	**1.02**†	**0.98**†	**0.98**†	**0.97**†

Table 5: Performance of THRec model with different side information on Movie dataset. † indicates significant improvement over all baseline methods (t-test, p-value < 0.05).

From the results we can see that the THRec method outperforms both THRec-tag and THRec-link methods significantly by 9.9% and 6.7%, respectively, when $k \leq 10$ and the side information is critical for the recommendation. Generally, there are two major reasons. First, compared to the genre tags (only 12 tags in total) and DBpedia entities (409 entities for $2,579$ movies, most of which are actors, directors, etc..), the user generated topics like "Special effect" and "War film" in topic hierarchies can better reflex the diversified and casual user interests. Second, topic hierarchy also performs more robust for unpopular or cold-start items due to the rich and timely social media contents within.

6 Conclusion

In this paper, we proposed a novel framework, i.e., the Topic Hierarchy based Recommendation model to tackle the problem of data sparseness in recommendation systems using items' relevant social media contents. In particular, we first converted the contents into topic hierarchy derived side information, including topic-item ratings, topic relatedness and topic popularity. Then we proposed an extended latent factor based recommendation model to optimize their impacts on

[10] http://wiki.dbpedia.org/Datasets

the recommendation results. The evaluation results demonstrated the superiority of our proposed model and the side information embedded in topic hierarchies.

In our future work, we will explore potential applications based on the THRec framework. It is also interesting to integrate other kinds of side information into our model.

Acknowledgements

This work was partly supported by the National Basic Research Program (973 Program) under grant No. 2012CB316301/2013CB329403, the National Science Foundation of China under grant No. 61332007, and the Tsinghua University Initiative Scientific Research Program under No. 20121088071.

References

[Cheng *et al.*, 2014] Jian Cheng, Ting Yuan, Jinqiao Wang, and Hanqing Lu. Group latent factor model for recommendation with multiple user behaviors. In *SIGIR'14*, pages 995–998, New York, NY, USA, 2014. ACM.

[Koren and Bell, 2011] Yehuda Koren and Robert Bell. Advances in collaborative filtering. In *Recommender Systems Handbook*, pages 145–186. Springer, 2011.

[Koren, 2008] Yehuda Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434. ACM, 2008.

[Leung *et al.*, 2011] Kenneth Wai-Ting Leung, Dik Lun Lee, and Wang-Chien Lee. Clr: a collaborative location recommendation framework based on co-clustering. In *SIGIR'11*, pages 305–314. ACM, 2011.

[McAuley and Leskovec, 2013] Julian McAuley and Jure Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 165–172. ACM, 2013.

[Ming *et al.*, 2010] Zhao-Yan Ming, Kai Wang, and Tat-Seng Chua. Prototype hierarchy based clustering for the categorization and navigation of web collections. In *SIGIR*, pages 2–9. ACM, 2010.

[Mnih and Salakhutdinov, 2007] Andriy Mnih and Ruslan Salakhutdinov. Probabilistic matrix factorization. In *Advances in neural information processing systems*, pages 1257–1264, 2007.

[Moshfeghi *et al.*, 2011] Yashar Moshfeghi, Benjamin Piwowarski, and Joemon M Jose. Handling data sparsity in collaborative filtering using emotion and semantic based features. In *SIGIR'11*, pages 625–634. ACM, 2011.

[Ostuni *et al.*, 2013] Vito Claudio Ostuni, Tommaso Di Noia, Eugenio Di Sciascio, and Roberto Mirizzi. Top-n recommendations from implicit feedback leveraging linked open data. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 85–92. ACM, 2013.

[Park *et al.*, 2013] Sunho Park, Yong-Deok Kim, and Seungjin Choi. Hierarchical bayesian matrix factorization with side information. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI '13, pages 1593–1599. AAAI Press, 2013.

[Popescul and Ungar, 2001] Rin Popescul and Lyle H. Ungar. Probabilistic models for unified collaborative and content-based recommendation in sparsedata environments. In *In UAI 01, 437C444*, 2001.

[Rafailidis *et al.*, 2014] Dimitrios Rafailidis, Apostolos Axenopoulos, Jonas Etzold, Stavroula Manolopoulou, and Petros Daras. Content-based tag propagation and tensor factorization for personalized item recommendation based on social tagging. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 3(4):26, 2014.

[Singh and Gordon, 2008] Ajit P Singh and Geoffrey J Gordon. Relational learning via collective matrix factorization. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 650–658. ACM, 2008.

[Su and Khoshgoftaar, 2009] Xiaoyuan Su and Taghi M. Khoshgoftaar. A survey of collaborative filtering techniques. *Adv. in Artif. Intell.*, 2009:4:2–4:2, January 2009.

[Tso-Sutter *et al.*, 2008] Karen HL Tso-Sutter, Leandro Balby Marinho, and Lars Schmidt-Thieme. Tag-aware recommender systems by fusion of collaborative filtering algorithms. In *Proceedings of the 2008 ACM symposium on Applied computing*, pages 1995–1999. ACM, 2008.

[Wang *et al.*, 2014] Jingjing Wang, Changsung Kang, Yi Chang, and Jiawei Han. A hierarchical dirichlet model for taxonomy expansion for search engines. In *Proceedings of the 23rd international conference on World wide web*, pages 961–970. International World Wide Web Conferences Steering Committee, 2014.

[Zhang *et al.*, 2013] Yongfeng Zhang, Min Zhang, Yiqun Liu, and Shaoping Ma. Improve collaborative filtering through bordered block diagonal form matrices. In *SIGIR'13*, pages 313–322. ACM, 2013.

[Zhu *et al.*, 2013] Xingwei Zhu, Zhao-Yan Ming, Xiaoyan Zhu, and Tat-Seng Chua. Topic hierarchy construction for the organization of multi-source user generated contents. In *SIGIR'13*, pages 233–242. ACM, 2013.

[Ziegler *et al.*, 2004] Cai-Nicolas Ziegler, Georg Lausen, and Lars Schmidt-Thieme. Taxonomy-driven computation of product recommendations. In *Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 406–415. ACM, 2004.