2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)
Banff Center, Banff, Canada, October 5-8, 2017

Fine-grained News Recommendation by Fusing Matrix Factorization, Topic Analysis and Knowledge Graph Representation

Kuai Zhang*, Xin Xin*, Pei Luo*, Ping Guo†*
*School of Computer and Technology
Beijing Institute of Technology, Beijing 100081, P.R. China
Email: kzhang@bit.edu.cn; xxin@bit.edu.cn; pluo@bit.edu.cn
†Image Processing and Pattern Recognition Laboratory
Beijing Normal University, Beijing 100875, P. R. China
Email: pguo@bnu.edu.cn

***Abstract*—Most news recommendation methods focus on using textual information of news to solve data sparseness problem of collaborative filtering. While if the text is not informative enough, these methods can't work well. A collaborative model combining matrix factorization, topic analysis and knowledge graph representation is proposed by introducing the knowledge from external knowledge base to alleviate the deficiency of the text. The experiment conducted on real life news dataset shows that the joint model outperforms the state-of-the-art method by 14% in Recall@200 metric, and improves the recommendation performance on sparse items by 20%.**

Keywords—news recommendation; collaborative model; knowledge graph; topic analysis

I. INTRODUCTION

Collaborative Filtering (CF) with textual information is used in text recommendation to solve the "cold start" and "data sparseness" problem in recommendation system. Topic models e.g. LDA[1], which simulate the generation of an article by a series of latent topic distributions to sample words in different topics, can extract knowledge from text. Under the assumption of topic model, the words that co-occurred frequently in corpus would be regarded in a same topic. However, if a word occurred only once or a few times in the total corpus, the traditional topic model couldn't extract information of these words generally. Table 1 shows a frequency of words that occurred in the news dataset which is collected from Hupu [1]from January to March in 2017. We can see that 87% of the words in the dictionary appeared less than 10 times; we call them "long tail" words. Due to that the most "long tail" words in the dictionary wouldn't be modeled by the traditional topic model, most fine-grained information of the news has been missed.

With the development of knowledge engineering, more and more structured information has been collected from the internet, Large scale knowledge bases such as Freebase, Wikipedia, DBpedia appeared, which contain millions of entities and depict the relations between them. By introducing the external knowledge base, the lack of knowledge of "long tail" words in the text would be made up.

Table 1 The words frequency on Hupu news

# words	28826
# frequency<=10	25285

This paper is inspired by the idea of using knowledge base to recommend news via extracting fine-grained features of text. Our contributions include the following two aspects:

- Design a novel model named Collaborative Entity Topic Ranking(CETR) combining the user behavior, topic analysis of plain text and representation of the knowledge graph. It introduces the information in the knowledge graph to make up the weakness of word-level topic model on "long tail" words and solves data sparseness problem in a fine-grained perspective.
- We evaluate our model in a real life dataset, and the experiment result shows that our model outperforms the state-of-the-art method by 14% in Recall@200.

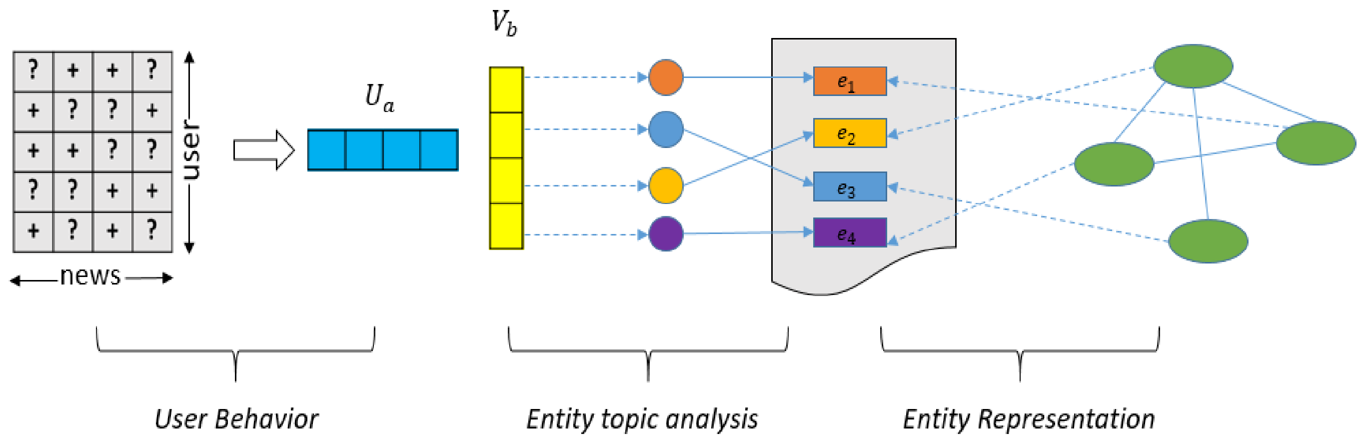
II. RELATED WORK

A. Fundamental Collaborative Filtering

The CF systems are required to combine the users and items to learn their properties collaboratively. The approach of CF can be divided into two main techniques: the neighborhood approaches and latent factor models. The neighborhood approaches concentrate on the relation between users or items. For example, the neighborhood approaches based on items[2,3] use a user's behavior on items to predict the user's behavior on other similar items. Better scalability and improved accuracy make the item-based approach more favorable in many cases[4,5]. The latent factor models, e.g. the latent semantic analysis[6], matrix factorization[7], map items and users into the same semantic space. Using the features on the semantic space

[1] https://www.hupu.com , a popular sport site in China.

Figure 1 The framework of Collaborative Topic Ranking (CETR)



can explain the observed behaviors between users and items. Many models are based on the latent factors models, such as the neural network models[8], SVD[9], SVD++[10]. Our model utilizes a ranking method based on matrix factorization called BPR[11] which could learn a divergence between users' likeness and unlikeness on items to model users' reading behaviors on news.

B. Recommender System with Textual Information

For the items with textual information, researchers utilize the knowledge from text to alleviate the weakness of data sparseness. topic models are used to extract the textual features of items[12,13] incorporated with the matrix factorization as a joint framework. For example, [14] views an item's feature vector as its topic proportion plus a bias which models the personalized interaction between a user and an item. Different from that, [15] proposes a softmax transfer function to connect the item's feature on user-item behaviors and the item's textual topics. In this paper, we propose an entity-level topic analysis model which extends from [15]. Because it combines the user behavior and the topic proportion more tightly via the transfer function.

C. Recommender System with Knowledge Base

The Knowledge bases which collect structured information from the internet are used to help to improve recommender system performance in recent years. [16] proposes an approach which extracts hierarchical knowledge information from DBpedia and then uses a graph algorithm to search the entity for recommendation.[17] recommends music for users by the semantic similarity between entities. [18] collects heterogeneous information from the knowledge base which is combined with matrix factorization to improve the accuracy of recommendation.

In our model, different from word-level topic analysis, we introduce an entity-level topic analysis to make up the weakness of word-level topic model which caused by the "long tail" words. In the aspect of using knowledge base information, we consider items as a cluster of entities and then analyze its semantic features on the entity level to learn more from the knowledge base, instead of simply regarding the item as an individual entity.

III. PROBLEM DEFINITION

As is shown in Figure 1, the users' reading behaviors on news can be modeled as a matrix. The position tagged with '+' indicates a user has read the news, and a position tagged with '?' indicates a user hasn't read this news. For every news, it contains several entities. For each entity of the news we can find a linked entity of the knowledge base. If provided the (1) user behaviors, (2) news contents and (3) the knowledge base, how to represent the knowledge information to connect the news' topic analysis with the knowledge base information. How does the topic analysis of news connected with the user behaviors ? How does the three parts work collaboratively?

IV. MODEL

Figure 1 depicts the whole framework of our news recommender system. The system consists of three parts: the first part models the users' behavior on news, the second part models the entities' generation in corpus, and the last part models the entities' representation in the knowledge graph. In this section, we will introduce the tree parts separately and then integrate them together as a joint model.

A. User Behavior

In this section, we introduce our method of modeling the users' reading behaviors on the news. A basic matrix factorization model can extract the user features and the news features in the same semantic space. In the paper, we deploy Bayesian Personalized Ranking (BPR) [11], which could learn the user vectors and news vectors by a ranking-based optimization objective.

First we define the parameters in this model, U is a $n \times k$ matrix, where n indicates the number of users and U_a denotes the k-dimensions latent feature vector of user a. V is an $m \times k$ matrix where m indicates the number of news and V_b denotes the k-dimensions latent feature vector of news b. A triplet (a, b, b') denotes that user a read news b but hasn't read news b', and we define $b \succ_a b'$ indicates that user a prefers news b than news b', then the BPR model uses a sigmoid function to describe the probability of observing the triplet (a, b, b'):

$$p(b >_a b'|U,V) = \frac{1}{1+e^{-(U_a^T V_b - U_a^T V_{b'})}}$$

Let D denotes all the triplets we could collect from the user behavior data, then the likelihood of observing all the triplets can be described as:

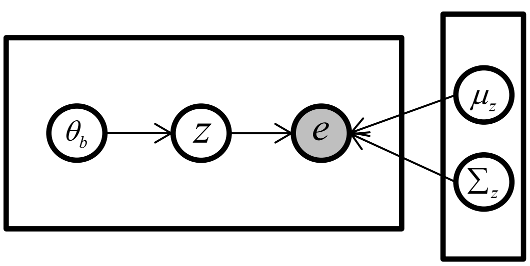
$$p(D|U,V) = \prod_{(a,b,b')\in D} p(b >_a b'|U,V)$$

Then we can learn U and V by maximum the likelihood function.

B. Entity Topic Analysis

In this section, we explain our model of making topic analysis of an article on the entity level. As we know, the entities and keywords could express the main idea of an article usually. So we make topic analysis on the generation of this entities in news, which is called entity level topic analysis. The entities and keywords could be represented as vectors according to their relationships with other entities in the knowledge base, which would be described in the next section. As is shown in the following picture, we assume that the entities in the same topic are sampled from a Gaussian distribution instead of a multinomial distribution compared with the word-level topic analysis.

Figure 2 Probability graph of entity level topic model



Let θ denotes a $m\times k$ matrix and θ_b denotes the topic proportion with k topics of news b, let Z_{bj} denotes the topic of the jth entity in news b. For each topic, a Gaussian distribution describes the distribution of the entities under the topic, μ_z and Σ_z denote the mean value the covariance of topic Z's Gaussian distribution N_z. Let e_{bj} denote the vector representation of the jth entity in news b.

Now let's describe the generation of entities in one news. The probability of observing entity e_{bj} would be $\theta_{z_{bj}} N_{z_{bj}}$ (given the topic assignment of each entity) and then the likelihood of observing the all entities in the corpus can be:

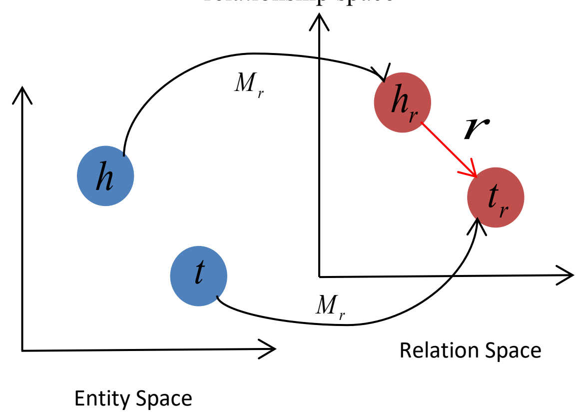
$$p(\Gamma|\theta,N) = \prod_{b\in\Gamma}\prod_{j}^{M_k} \theta_{z_{bj}} N_{z_{bj}}$$

Where Γ denotes the set of news and M_b denotes the number of entities in news b. A Gibbs Sampling [19] method is usually used to learn the parameters of above likelihood.

C. Entity Representation

In our model, we want to represent the entities into a continuous space so that we can connect the knowledge base information with topic analysis. The knowledge base stores heterogeneous information and rich relationships between entities. TransE [20] proposes a promising approach which defines the two entity e_h and e_t related by relation r should satisfy the equation: $v_{e_h} + v_r = v_{e_t}$. It projects the entity vector and the relation vector into the same continuous space. However this approach couldn't represent relations in the form of one-to-many, TransR[21] brings in a relation projection matrix and solved this problem. Hence, we use TransR as the method to represent the entities in the knowledge base.

Figure 3 Projecting entity vector from entity space to relationship space



As is shown in Figure 3, assuming the head entity h and the tail entity t are related with the relation r. For each relation r, a projection matrix is defined with it to project entities from entity space into relation space.

v_h and v_t are k-dimensions vectors which denote the representation of entity h and entity t in the entity space. m_r is a d-dimensions vector which denotes the representation of relation r. M_r is a $k\times d$ matrix which projects v_h and v_t to the relationship space of r. v_{h_r} and v_{t_r} denote the representation of h and t in the space of r. From the assumption of TransR, the following equations should be satisfied:

$$v_{h_r} = v_h M_r \qquad v_{t_r} = v_t M_r$$

$$v_{h_r} + m_r = v_{t_r}$$

Similar to BPR, we define the probability of observing the quadruple (*h, r, t, t'*) which means entity h has a relation r with t but doesn't have a relation with t':

$$p(t >_{h,r} t'|v,M,m) = \frac{1}{1+e^{-(||v_h M_r + m_r - v_{t'} M_r||_2^2 - ||v_h M_r + m_r - v_t M_r||_2^2)}}$$

Then the likelihood of observing all quadruple E in the knowledge base is:

$$p(E|v, M, m) = \prod_{(h,r,t,t')\in E} p(t \succ_{h,r} t'|v, M, m)$$

D. Joint Model

In this section, we combine the above three models together so that the three parts can help each other to improve the effect of the recommendation system.

First, we define the parameters, U is a $n\times k$ matrix denotes the user features, V is an $m\times k$ matrix denotes the news features, θ is also an $m\times k$ matrix denotes the topic proportion of news. μ_z and Σ_z denote the mean value and the variance of the Gaussian distribution of topic z. $z_{d,k}$ denotes the topic of the kth entity in news d. E is a $p\times q$ matrix denotes the entities' representation. p is the size of the entity set and q is the dimension of each entity vector. R is a $w\times u$ matrix denotes the representation of relationships in knowledge base. w is the size of relationships in knowledge base and u is the dimension of a relation vector. For each relation r, M_r is a $q\times u$ matrix that projects the vector in entity space to the relationship space. In this model, we define $\Theta = \{U, V, \theta, \mu, \Sigma, E, R, M\}$.

Given all triplets $(a, b, b') \in D$, topic assignment of each entity in all news Γ and all quadruples $(h, r, t, t') \in$ S.

The triplets mean that user a read news b but hasn't read news b' and the quadruples mean that entity h and entity t have relation r but h and t' don't have the relation r in the knowledge base. The log likelihood of observing all the triplets, entities and quadruples is:

$$\begin{aligned} p(D, S, \Gamma|\Theta) = &\sum_{(a,b,b')\in D} ln\sigma(U_a^T V_b - U_a^T V_{b'}) \\ &- \lambda_1 \sum_{d\in\Gamma}\sum_{k=1}^{n} ln\theta_{dz_{d,k}} N(E_{dk}; \mu_{z_{d,k}}, \Sigma_{z_{d,k}}) \\ &- \lambda_2 \sum_{(h,r,t,t')\in S} ln\sigma(||E_h M_r + R_r \\ &- E_{t'} M_r||_2^2 - ||E_h M_r + R_r - E_t M_r||_2^2) \end{aligned}$$

Where σ denotes the sigmoid function, λ_1 and λ_2 denote the weight for topic analysis and entity representation.

We borrow the idea of [15] to set $\theta_{i,k} = \frac{e^{V_{i,k}}}{\sum_{k'} e^{V_{i,k'}}}$ to bridge the user behavior part and the topic analysis part through the softmax function, which make θ_i a probability distribution satisfied $\sum_k \theta_{i,k} = 1$.

PARAMETER FITTING

We fit these parameters which are described above by a sampling-like method, which consists of two steps:

a) For each entity e.g. E_{dj} in the news corpus, sample the topic with the probability :

$$p\left(z_{dj}^{(t)} = k\right) = \theta_{z_{dj}}^{(t)} N(E_{dj}^{(t)}; \mu_{z_{dj}}^{(t)}, \Sigma_{z_{dj}}^{(t)})$$

b) Update the $\Theta^{(t+1)}$ given the topic assignment by minimum $-p(D, S, \Gamma|\Theta^{(t)})$ with gradient descent method.

Repeat *a)* and *b)* until convergence, the experiment proves that this optimization approach will converge in about 10 iterations on our dataset.

V. EXPERIMENT

A. DataSet

Table 1 Detail statistics on dataset

	user	news
Min. #feedback	5	5
Max. #feedback	126	387
Avg. #word	-	81.26
Total. #word	-	786922
Avg. #entity	-	11.53
Total. #entity	-	111656

News dataset. In order to exploit fine-grained interests of users, we crawl news data from **Hupu**[2], a sports portal in China. The crawled news dataset has 11078 news, 5239 users and 167431 ratings in total. We remove the user/news that read/be read less than 5 times, then get our final dataset with 9684 news and 3118 users with 132713 ratings, the sparseness on the dataset is 99.6%.

Knowledge base. We choose the free knowledge base **Wikidata**[3], which contains 25672886 entities, as our experiment knowledge base. Only the entities which has Chinese label or alias and the relationship between these entities are selected. Finally, 2025669 entities and 8767366 relationships and 735 kinds of relations are extracted. Table 1 shows details about the news dataset.

B. Experiment Setup

1) DataSet Allocation and the Metric

We divide our dataset into a training dataset and a testing dataset by random selection with a proportion of 8:2, for all the algorithms mentioned following, using the same training dataset and testing dataset.

To evaluate the performance of each model, we introduce the Recall@K metric, it can be calculated by:

$$Recall@K = \frac{number\ of\ \ news\ the\ user\ likes\ in\ top\ K}{total\ number\ of\ news\ the\ user\ likes}$$

The denominator denotes the top K news ranked by the score function of models, e.g. in BPR. We set $Score(u, v) = u^T v$, where u denotes the feature vector of the user and v denotes the feature vector of the news. For achieving the Recall@K on the whole dataset, we calculate the Recall@K of each user in the test dataset and then calculate the average value of these Recall@K values.

2) Baselines and CETR Model

[2] https://www.hupu.com
[3] https://www.wikidata.org

Table 3 Overall performance of baselines and CETR

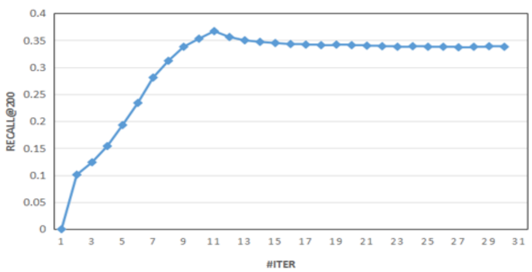
Train	Popurlarity	BPR	HFT	BPR+ELDA+KBR	HFT+KBR	CETR	Imp.1	Imp.2
0	0.2078	0.3001	0.3201	0.3426	0.3525	**0.3673**	14.70%	4.20%
1	0.2069	0.3037	0.3203	0.3408	0.3531	**0.3672**	14.60%	4.00%
2	0.2045	0.3003	0.3189	0.3433	0.3541	**0.3682**	15.50%	4.00%
3	0.2068	0.3197	0.3204	0.3395	0.3544	**0.3653**	14.00%	3.10%
4	0.2051	0.3001	0.3144	0.3482	0.3572	**0.3657**	16.30%	2.40%
5	0.2063	0.2998	0.3197	0.3403	0.3532	**0.3621**	13.30%	2.50%
6	0.2017	0.2996	0.3206	0.3491	0.3516	**0.3624**	13.00%	3.10%
7	0.2053	0.3003	0.3183	0.3403	0.3521	**0.3613**	13.10%	2.60%
Avg	0.2055	0.3029	0.3191	0.343	0.3535	**0.3649**	14.30%	3.20%

Table 4 Topic words on word-level and entity-level

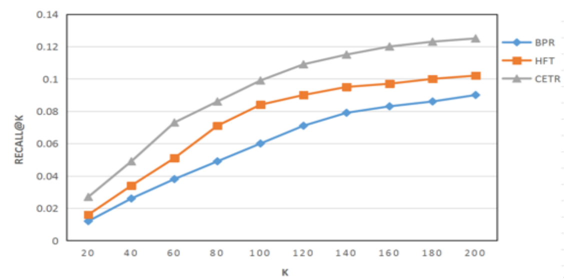
Topic on word level			
Soccer	Media	Basketball	Competition
Bassa	report	Warriors	play
football team	international	assists	score
club	wechat	Rockets	defend
competition season	news	NBA	backbord

Topic on entity level			
Basketball Player	NBA Team	European Football Club	Chinese football
Dell Curry	Celtics	Southampton F.C.	Jin Taiyan
George Best	Magic	Real Madrid FC	Guangzhou Evergrande
Smush Parker	Rockets	Nottingham Forest F.C.	Beijing Guoan F.C.
Shaquille O'Neal	Raptors	Middlesbrough F.C.	Chinese Super League

Figure 4 (a) Convergence of CETR



(b) Performance on sparse news



In order to show how well our model works, we choose several baselines to compare with it. These baselines include: ***1)*** **Popularity.** This method simply recommend user by the news which are read by most users.***2)*** **B**ayesian **P**ersonalized **R**anking (**BPR*)***. This method is introduced in the model section of this paper, which only user the user reading record to make recommendation. ***3)*** **H**idden **F**actors as **T**opics (**HFT*)***. This algorithm is proposed by [15], which uses the textual inf ormation by combining the topic analysis of articles and the user behaviors. To show how well the collaborative learning of user behavior, topic analysis and knowledge base representation, we divided our model by ***1)*** **BPR+ETA+KBR.** This method first learn the entity representation by TransR[22] which we call **K**nowledge **B**ase **R**epresentation (**KBR**). In next step, we use the entity representation to analyze the topic proportion of article in entity level which we call **E**ntity **T**opic **A**nalysis (**ETA**). Finally train **BPR** by fixing the news vectors with the topic proportions of news learned by **ETA**, ***2)*** **HFT+KBR**. We want to know how the collaborative learning of entity representation influences the recommendation performance. First we learn the representation of entities in knowledge base, and then we learn the user behavior part and topic analysis part collaboratively. We introduce above methods to be compared with our method which called **C**ollaborative **E**ntity **T**opic **R**anking (**CETR**).

3) Hyper-parameters Setting

We choose the hyper-parameters by cross-validation method, for different model, the hyper-parameters could be various. For instance, we choose $k=10$ for HFT model and choose $k=20$ for our model where k denotes the number of topic proportions. Finally, we set λ_1=0.1 $\lambda_2 = 10$, k=20, and $n = 10$ where λ_1 , λ_2 denote the weight of the topic analysis part and knowledge base representation part. n denotes the dimension of the entity vector.

4) Entiies Extraction

We extract entities in every news to introduce the information of knowledge base for topic analysis. The entities of news are divided into two classes: one is the key words of the news which ranked by the TF-IDF value and top 5 are selected for each document; another class of entities is the named entities recognized by a NER toolkit[4]. We use the Wikidata API[5] for linking entity to Wikidata by choosing the top 1 entity in the search result.

[4] https://github.com/HIT-SCIR/ltp
[5] https://www.wikidata.org/w/api.php

C. Overall Performance and Convergence

We randomly generate eight different training sets and testing sets to verify the stability of our model and for each dataset we compared our model with the baselines. Experiment results are showed in Table 3.

We select recall@200 as our metric and the overall performance of our model is around 0.36. Imp.1 denotes the improvement that our model outperforms HFT and Imp.2 de notes the improvement that our model outperforms HFT+KBR.

To verify the convergence of our model, we record the recall@200 of each iteration in total 30 iterations. Figure 4(a) shows our model converged in around 10 iterations.

D. Topic Analysis

We select some typical topic words on both word-level topic analysis and entity-level topic analysis which are showed in Table 4. It can be viewed that the entity-level topic words express the topics of an article in a fine-grained perspective, even focusing on the football club rather than cluster some high-frequency co-occurrence words in a rough topic such as basketball. The detail of an article could be expressed by entity as well as the detail of the user's interests.

E. Performance on Long-Tail News

In order to show the recommendation performance of CETR for news that are read by only a few users, which are called "long-tail" news, we select news that read by less than 10 users in our dataset. Then we choose 20% of user reading records on these "long-tail" news as testing set (10095 reading records), and the remaining data (122618 reading records) as training set. Comparison of CETR, HFT and BPR on Recall@K is showed in Figure 4(b). Our model outperforms HFT by around 20%, which indicates the data sparseness problem is alleviated by introducing knowledge base information.

VI. Conclusion

We propose a joint model to learn the user behavior, topic analysis and knowledge base representation collaboratively to improve the performance of news recommendation. The experiment results show that recommendation performance is improved by 14% on Recall@200 metric by introducing external knowledge.

ACKNOWLEDGMENT

This work is fully supported by the grants from the National Natural Science Foundation of China (61672100, 61375045), the Joint Research Fund in Astronomy (U1531242) under cooperative agreement between the National Natural Science Foundation of China (NSFC) and Chinese Academy of Sciences (CAS), Beijing Natural Science Foundation (4162054). Associate Prof. Xin Xin is the author to whom all the correspondence should be addressed.

References

[1] Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation[J]. Journal of Machine Learning Research, 2003, 3:993-1022.

[2] Linden G, Smith B, York J. Amazon.com Recommendations: Item-to-Item Collaborative Filtering[J]. IEEE Internet Computing, 2003, 7(1):76-80.

[3] Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms[C]// International Conference on World Wide Web. ACM, 2001:285-295.

[4] Linden G, Smith B, York J. Amazon.com Recommendations: Item-to-Item Collaborative Filtering[J]. IEEE Internet Computing, 2003, 7(1):76-80.

[5] Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms[C]// International Conference on World Wide Web. ACM, 2001:285-295

[6] Hofmann T. Latent semantic models for collaborative filtering[J]. Acm Transactions on Information Systems, 2004, 22(1):89-115.

[7] Tak, Cs G, Szy I, et al. Matrix factorization and neighbor based algorithms for the netflix prize problem[C]// ACM Conference on Recommender Systems, Recsys 2008, Lausanne, Switzerland, October. DBLP, 2008:267-274.

[8] Salakhutdinov R, Mnih A, Hinton G. Restricted Boltzmann machines for collaborative filtering[C]// International Conference on Machine Learning. ACM, 2007:791-798.

[9] Deerwester B S, Dumais S F, Landauer G, et al. Indexing by latent semaintic analysis[C]// JASIS. 1990.

[10] Koren Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2008:426-434.

[11] Rendle, Steffen, Freudenthaler, et al. BPR: Bayesian personalized ranking from implicit feedback[J]. 2012:452-461.

[12] Diao Q, Qiu M, Wu C Y, et al. Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS)[J]. 2014:193-202.

[13] Agarwal D, Chen B C. fLDA:matrix factorization through latent dirichlet allocation[C]// International Conference on Web Search and Web Data Mining, WSDM 2010, New York, Ny, Usa, February. DBLP, 2010:91-100.

[14] Wang C, Blei D M. Collaborative topic modeling for recommending scientific articles[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, Ca, Usa, August. DBLP, 2011:448-456.

[15] Mcauley J, Leskovec J. Hidden factors and hidden topics: understanding rating dimensions with review text[C]// ACM Conference on Recommender Systems. ACM, 2013:165-172.

[16] Cheekula S K, Kapanipathi P, Doran D, et al. Entity recommendations using hierarchical knowledge bases[J]. Ceur Workshop Proceedings, 2015.

[17] Passant A. dbrec: music recommendations using DBpedia[C]// The Semantic Web - ISWC 2010 -, International Semantic Web Conference, ISWC 2010, Shanghai, China, November 7-11, 2010, Revised Selected Papers. DBLP, 2010:209-224.

[18] Zhang F, Yuan N J, Lian D, et al. Collaborative Knowledge Base Embedding for Recommender Systems[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2016:353-362.

[19] Gelfand A E. Gibbs Sampling[J]. Journal of the American Statistical Association, 2000, 95(452):1300-1304.

[20] Bordes A, Usunier N, Garcia-Duran A, et al. Translating Embeddings for Modeling Multi-relational Data[J]. Advances in Neural Information Processing Systems, 2013:2787-2795.

[21] Lin Y, Liu Z, Sun M, et al. Learning entity and relation embeddings for knowledge graph completion[C]// Twenty-Ninth AAAI Conference on Artificial Intelligence. AAAI Press, 2015:2181-2187.