Exploring Review Content for Recommendation via Latent Factor Model

Xiaoyu Chen[1,2], Yuan Yao[1,2], Feng Xu[1,2], and Jian Lu[1,2]

[1] State Key Laboratory for Novel Software Technology, Nanjing, China
[2] Department of Computer Science and Technology, Nanjing University, China
{chenxiaoyu,yyao}@smail.nju.edu.cn, {xf,lj}@nju.edu.cn

Abstract. Recommender systems have been widely studied and applied in many real applications such as e-commerce sites, product review sites, and mobile App stores. In these applications, users can provide their feedback towards the items in the form of ratings, and they usually accompany the feedback with a few words (i.e., review content) to justify their ratings. Such review content may contain rich information about user tastes and item characteristics. However, existing recommendation methods (e.g., collaborative filtering) mainly make use of the historical ratings while ignore the content information. In this paper, we propose to explore the review content for better recommendation via latent factor model. In particular, we propose two strategies to leverage the review content. The first strategy incorporates review content as a guidance term to guide the learnt latent factors of user preferences; the second strategy formulates a regularization term to constrain the preference differences between similar users. Experimental evaluations on two real data sets demonstrate the usefulness of review content and the effectiveness of the proposed method for recommendation.

Keywords: Recommender system, latent factor model, review content, guidance term, regularization term.

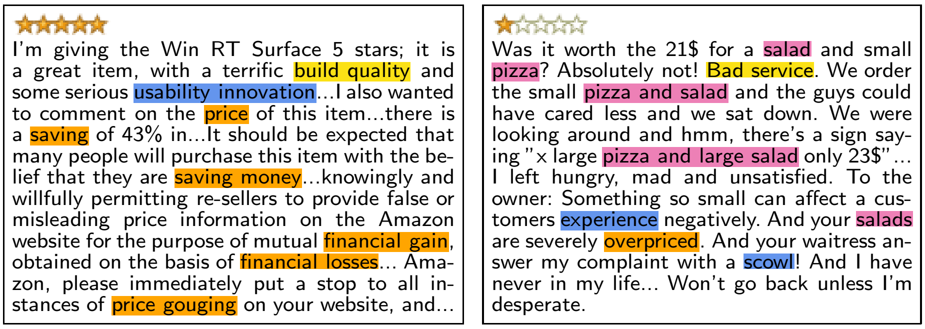
1 Introduction

Recommender systems become indispensable in many real applications such as Netfilx[1], Amazon[2], last.fm[3], etc. In these applications, users may give ratings on items, and the given ratings may be in turn used to predict the preferences of users for unknown items. In addition to the ratings, users usually accompany them with a few words (i.e., review content) to justify their ratings. Such review content may contain rich information about user preferences and item characteristics.

Consider two sample reviews from Amazon and Yelp as shown in Figure 1. While the rating (1 star or 5 stars) indicates user's overall satisfaction on the

[1] http://www.netflix.com
[2] http://www.amazon.com
[3] http://www.last.fm

D.-N. Pham and S.-B. Park (Eds.): PRICAI 2014, LNAI 8862, pp. 668–679, 2014.



(a) Review of Surface RT from Amazon (b) Review of a restaurant from Yelp

Fig. 1. Two sample reviews from Amazon and Yelp

items, the review content contains more fine-grained information. For example, the user in the left figure focuses on the price aspect of the item while other aspects such as usability and quality are also mentioned; the user in the right figure mainly complains the service and the price aspects.

Despite the rich information in review content, existing recommendation approaches (e.g., collaborative filtering) tend to ignore such information due to the following challenges. First, review content cannot be directly used as features under the supervised machine learning framework, because the review content is unavailable for the user-item pair that is being predicted. Second, it is unclear how to incorporate the review content into the latent factor model, especially when the review content is usually noisy.

In this paper, we propose to explore the review content to improve recommendation accuracy. To address the above challenges, we first aggregate the review content along users and items. For example, on the user side, we aggregate the reviews from a given user altogether and treat them as a document for this user. Similar documents can be generated for each item. Next, we propose two strategies to incorporate the review content into the latent factor model. The first strategy treats the review content as prior knowledge and incorporates it as a guidance term. In particular, we apply the LDA (Latent Dirichlet Allocation) [1] model to obtain some latent topics and use these topics to guide the latent factor learning. The second strategy formulates a regularization term to constrain the preference differences between similar users. The similarities between users are computed based on user documents. Since the *nouns* (e.g., price, service) in user/item documents are less noisy and they may better reflect the aspects that are of interest to the users, we apply the LDA model on the nouns only to identify the preference aspects of users. Experimental evaluations on two real data sets demonstrate the usefulness of review content and the effectiveness of the proposed method for recommendation. For example, with deliberate treatment on the review content, the prediction accuracy can be improved by up to 12.3%.

The remainder of this paper is organized as follows. Section 2 overviews related work. Section 3 describes the proposed method. Section 4 presents the experimental evaluations. Section 5 concludes the paper.

2 Related Work

In this section, we briefly review related work including traditional recommender systems and the existing studies on review content in recommender systems.

A major body of traditional recommendation approaches focus on the user-item rating matrix, and these methods are widely adopted in commercial recommender systems [2]. These methods can be divided to memory-based and model-based approaches. The base idea of memory-based methods is to find similar users or items for making the prediction filtering [3,4]. On the other hand, model-based methods employ statistical and machine learning techniques to learn models from the user-item rating matrix [5,6]. In the past few years, the matrix factorization method or latent factor model [7,8] has become the most prevailing method due to its flexibility. However, all the above methods ignore the review content, which might be of great importance to the recommendation setting.

Recently, many researchers have started to analyze the review content. For example, Moghaddam et al. [9,10] propose to identify different aspects of an item in both item level and review level. Wang et al. [11,12] aim to discover reviewer's latent opinion on each aspect in both supervised setting and unsupervised setting. Sentiment analysis is also applied on review content [13,14]. Different from the above works which focus on the analysis and aspect identification of review content, our goal is to make use of the review content to help recommendations.

The most relevant work is from Agarwal et al. [15] and McAuley et al. [16]. They also aim to improve the prediction performance by combining the review content with ratings. The focus of these two methods is on the item side (by aggregating the reviews along items), while we focus on the user side. Therefore, our method could be potentially complementary to these two methods.

3 Methods

In this section, we describe the proposed method. After we introduce some preliminaries, we will present the proposed strategies to leverage review content.

3.1 Preliminaries

Given the rating matrix $R \in \mathbb{R}^{m \times n}$ and review matrix C from m users to n items. User $u \in U$ may give rating r_{ui} along with review content c_{ui} to item i. A traditional latent factor model would factorize R into a user-specific matrix and an item-specific matrix [7,8]:

$$\hat{R} = P^T Q, \tag{1}$$

where $P \in \mathbb{R}^{f \times m}$ denotes the user latent factor vector, and $Q \in \mathbb{R}^{f \times n}$ is the item latent factor vector, f denotes the number of latent aspects. Typically, the following minimization problem is formulated to obtain the matrix P and Q:

$$\sum_{u=1}^{m} \sum_{i=1}^{n} I_{ui}(R_{ui} - P_u^T Q_i)^2 + \lambda(\|P\|_F^2 + \|Q\|_F^2), \tag{2}$$

where P_u/Q_i indicates the u^{th}/i^{th} column of P/Q, I_{ui} indicates whether user u has rated item i, and $\|\cdot\|_F^2$ denotes the Frobenius norm. Two regularization terms are added to avoid over-fitting. The parameter λ controls the degree of regularization and it is usually determined by cross-validation.

3.2 Processing Review Content

As indicated by the sample reviews in Figure 1, the topic distribution of reviews might be useful in identifying user preferences. However, existing topic models (e.g., Probabilistic Latent Semantic Analysis and Latent Dirichlet Allocation) use the *bag-of-words* representation for review content, which may cause them to suffer from the noise in the content. For example, whether the item is *cheap* or *costly*, the aspect that the user cares about is *price*. Therefore, we only keep the *nouns* in the review content.

Next, to make use of the review content, we aggregate the reviews along both user side and item side. Take user side as an example. For each user, we group all his previous reviews, keep the nouns, and treat it as a document for the user. By applying LDA on these documents, we can have the topic distribution $Y \in \mathbb{R}^{f\times m}$ which is relevant to the user preferences. Similarly, we can obtain the topic distribution in the item side as $Z \in \mathbb{R}^{f\times n}$.

Based on these processing steps, we will introduce the two proposed strategies in the next two subsections, respectively.

3.3 The Proposed Model with Guidance Term

Intuitively, the favor of user u on item i over f aspects is composed of two parts: the user's own preference and community's overall assessment of this item. Here, the own preference of user u can be considered as the Y_u because Y_u is obtained based on the document of user u. Similarly, the overall assessment can be considered as Z_i because Z_i is obtained from the document of item i. Then, we can add our guidance term for the prediction as

$$\hat{r}_{ui} = (\beta Y_u + \beta' Z_i + P_u)^T Q_i, \tag{3}$$

where $\beta, \beta' \geq 0$ control the importance of the two parts on this factor model. Further, the importance of the two parts should be related to the document size. Therefore, we add the following constraints on β and β':

$$\begin{cases} \beta + \beta' = \varphi_C \\ \frac{\beta}{\beta'} = \mu \frac{\|\mathcal{D}_u^{(U)}\|_L}{\|\mathcal{D}_i^{(I)}\|_L}, \end{cases} \tag{4}$$

where $\mathcal{D}_u^{(U)}$ indicates the document for user u, $\mathcal{D}_i^{(I)}$ indicates the document for item i, $\|\cdot\|_L$ denotes the amount of reviews in the document, parameter φ_C indicates the overall ratio of the guidance term, and parameter μ adjusts the weight between user's own preference and community's overall assessment. Here, we assume the amount of information in review content is positively correlated to the length of the content. For example, if the length of $\mathcal{D}_u^{(U)}$ is larger than that of $\mathcal{D}_i^{(I)}$, β deserves more weight in the guidance term.

By incorporating the guidance term into Eq. (2), we have the objective function of our first model,

$$\begin{aligned}\min_{P,Q} \mathcal{L}(R,P,Q) = &\sum_{u=1}^{m}\sum_{i=1}^{n} I_{ui}(R_{ui} - (\beta(u,i)Y_u + \beta'(u,i)Z_i + P_u)^T Q_i)^2 \\ &+ \lambda(\|P\|_F^2 + \|Q\|_F^2).\end{aligned} \tag{5}$$

To minimize the above formulation, a local minimum can be found by adopting gradient descent in P_u and Q_i,

$$\begin{aligned}\frac{1}{2}\cdot\frac{\partial\mathcal{L}}{\partial P_u} &= \sum_{i=1}^{n} I_{ui}((\beta(u,i)Y_u + \beta'(u,i)Z_i + P_u)^T Q_i - R_{ui})Q_i + \lambda P_u \\ \frac{1}{2}\cdot\frac{\partial\mathcal{L}}{\partial Q_i} &= \sum_{u=1}^{m} I_{ui}((\beta(u,i)Y_u + \beta'(u,i)Z_i + P_u)^T Q_i - R_{ui}) \\ &\quad\cdot\ (\beta(u,i)Y_u + \beta'(u,i)Z_i + P_u) \\ &\quad + \lambda Q_i.\end{aligned} \tag{6}$$

3.4 The Proposed Model with Regularization Term

In the previous subsection, we incorporate review content as a guidance term to guide the learnt latent factors of user preferences. In this subsection, we present the second strategy of formulating a regularization term to constrain the preference differences between similar users.

Intuitively, if user u and l have similar topic distributions, their preferences (P_u and P_l) should also be similar to each other. As indicated by Ma et al. [17], the following regularization term performs relatively well for recommendation:

$$\sum_{u=1}^{m}\sum_{l\in S(u)} sim(u,l)\|P_u - P_l\|_F^2, \tag{7}$$

where $sim(u,l)$ is the similarity between $\mathcal{D}_u^{(U)}$ and $\mathcal{D}_l^{(U)}$ over topics. we will discuss the similarity metric later in Section 3.4.

Then, by incorporating the above regularization term into Eq. (2), the objective function can be formulated as:

$$\begin{aligned}\min_{P,Q} \mathcal{L}(R,P,Q) = &\sum_{u=1}^{m}\sum_{i=1}^{n} I_{ui}(R_{ui} - P_u^T Q_i)^2 + \lambda_1(\|P\|_F^2 + \|Q\|_F^2) \\ &+ \lambda_2\sum_{u=1}^{m}\sum_{l\in S(u)} sim(u,l)\|P_u - P_l\|_F^2,\end{aligned} \tag{8}$$

where $S(u)$ is the user set which contains the users that are similar to user u.

Similar to the first model, by performing gradient descent in P_u and Q_i, a local minimum of the Eq. (8) can also be found,

$$\frac{1}{2} \cdot \frac{\partial \mathcal{L}}{\partial P_u} = \sum_{i=1}^{n} I_{ui}(P_u^T Q_i - R_{ui})Q_i + \lambda_1 P_u + \lambda_2 \sum_{l \in S(u)} sim(u,l)(P_u - P_l),$$
$$\frac{1}{2} \cdot \frac{\partial \mathcal{L}}{\partial Q_i} = \sum_{u=1}^{m} I_{ui}(P_u^T Q_i - R_{ui})P_i + \lambda_1 Q_i. \tag{9}$$

Similarity between Documents. For the similarity between documents, the intuitive words-with-frequency similarity is not adequate to this problem. When describing the experience people may use different terms to refer to the same topic (e.g., *value* and *money* for price). Since we have the preference vector $Y_u \in \mathbb{R}^{f \times 1}$ obtained from $\mathcal{D}_u^{(U)}$, the similarity between two users can be calculated by measuring the distance of the preference vectors. We adopt the cosine similarity to calculate the distance of the preference vector,

$$sim(u,l) = \frac{\sum_{k=1}^{f} Y_{uk} \cdot Y_{lk}}{\sqrt{\sum_{k=1}^{f}(Y_{uk})^2} \cdot \sqrt{\sum_{k=1}^{f}(Y_{lk})^2}}. \tag{10}$$

We only involve those users who are similar to the given user above a certain threshold,

$$S(u) = \{v | v \in U, sim(u,v) > \varphi_S\}. \tag{11}$$

3.5 Putting Everything Together: The Proposed GTRT Model

Finally, we present the proposed model (i.e., **GTRT**) by combining the guidance term and the regularization term. It should be pointed out that the length of $S(u)$ in Eq. (11) usually differs greatly due to the size of $\|\mathcal{D}_u^{(U)}\|_L$ and individual writing styles. Thus, we change the regularization term to

$$\lambda_2 \sum_{u=1}^{m} |S(u)|^{-\frac{1}{2}} \sum_{l \in S(u)} sim(u,l)\|P_u - P_l\|_F^2. \tag{12}$$

By putting the guidance term and the regularization term together, we have the following objective function,

$$\begin{aligned}\min_{P,Q} \mathcal{L}(R,P,Q) = &\sum_{u=1}^{m}\sum_{i=1}^{n} I_{ui}(R_{ui} - (\beta(u,i)Y_u + \beta'(u,i)Z_i + P_u)^T Q_i)^2 \\ &+ \lambda_1(\|P\|_F^2 + \|Q\|_F^2) \\ &+ \lambda_2 \sum_{u=1}^{m} |S(u)|^{-\frac{1}{2}} \sum_{l \in S(u)} sim(u,l)\|P_u - P_l\|_F^2. \end{aligned} \tag{13}$$

where the partial derivatives are shown as follows.

$$\begin{aligned}\frac{1}{2}\cdot\frac{\partial\mathcal{L}}{\partial P_u} &= \sum_{i=1}^{n} I_{ui}((\beta(u,i)Y_u+\beta'(u,i)Z_i+P_u)^T Q_i - R_{ui})Q_i + \lambda_1 P_u\\ &\quad + \lambda_2|S(u)|^{-\frac{1}{2}}\sum_{l\in S(u)} sim(u,l)(P_u-P_l),\\ \frac{1}{2}\cdot\frac{\partial\mathcal{L}}{\partial Q_i} &= \sum_{u=1}^{m} I_{ui}((\beta(u,i)Y_u+\beta'(u,i)Z_i+P_u)^T Q_i - R_{ui})\\ &\quad \cdot\ (\beta(u,i)Y_u+\beta'(u,i)Z_i+P_u)\\ &\quad + \lambda_1 Q_i. \end{aligned} \tag{14}$$

4 Experimental Evaluations

In this section, we present the experimental evaluations. The experiments are designed to answer the following questions. Is it useful to incorporate the review content for recommendation? To what degree can the review content help? What is the effect of different parameter choices? How does the proposed method perform under the cold-start scenarios?

4.1 Experimental Setup

Data Sets. Here, we first describe the data sets we used. The first data set is from Amazon reviews [18]. Amazon.com is considered as one of the most successful e-commerce sites, and much attention has been attracted to analyze the reviews in Amazon. There are 5.8 million reviews, 2.14 users and 6.7 million items in this data set. The items are categorized into *Books*, *Music*, *DVD/VHS* and *mProducts* (industry manufactured items like electronics, sundries). The *mProducts* category is often studied because reviews in the other three categories are more like descriptions instead of opinions. Moreover, we remove some inactive users who has rated less than 3 reviews, resulting 55,086 reviews and ratings from 11,011 users to 36,222 items.

Another data set is from the RecSys Challenge 2013[4]. The theme of this competition is personalized business recommendations for Yelp users. This Yelp data set provides a detailed snapshot over 200,000 reviews from the Phoenix, AZ metropolitan area. Similar to the first data set, we discard some inactive users and get 173,586 reviews and ratings from 23,890 users to 6,265 items. We also observed that the average review length from Yelp is longer than that from Amazon.

Evaluation Metrics. For evaluation metrics, we use the popular Root Mean Square Error (RMSE) metric to measure the prediction quality of our proposed approach. RMSE is defined as

[4] `http://www.kaggle.com/c/yelp-recsys-2013`

$$RMSE = \sqrt{\frac{1}{T}\sum_{u,i}(R_{ui} - \hat{R}_{ui})^2}, \tag{15}$$

where R_{ui} denotes the rating user u give to item i, $\hat{R}_{ui}$ denotes the rating user u give to item i as predicted by our approach, and T is the number of ratings in the test set.

Compared Methods. In our experiments, we compare the effectiveness of the following methods:

– **MEAN:** this is the baseline method which simply takes the average rating of a given user as this user's predictions.
– **LFM:** the Latent Factor Model is the standard matrix factorization method as mentioned in Section 3.1. Only ratings are used in this method.
– **GTM:** this is the method as shown in Section 3.3. It extracts topics from user reviews as guidance term for user preference learning.
– **RTM:** this method is shown in Section 3.4. It formulates a regularization term from user reviews and uses this terms as constraints.
– **GTRT:** the proposed method as shown in Section 3.5. This method combines GTM and RTM.

We divide both data sets into training set and test set. Specially, we randomly select 10% and 20% ratings as the test set and use the rest as training set. For the number of aspects f, we set it to 50.

4.2 Experimental Results

Review Aspect Identification. Here, we first show that the processing step can potentially improve the identification of review aspects. Recall that we assume the *nouns* are more valuable in aspect identification. Some topics and their top score words are shown in Table 1. The table shows both the cases when we use nouns only and when we use all the words. The results are from the Amazon data set after we apply the LDA model. We can see that topic words like *disagree, discuss, strong* from all words are relatively meaningless for extracting aspects, while topic words from *nouns* only are more informative for identifying latent aspects.

Rating Prediction. Next, we present effectiveness comparisons. Table 2 shows the RMSE results of the compared five methods on the two data sets. First, We can see from the table that all the three methods that employ the review content (i.e., GTM, RTM and GTRT) gain a significant improvement over the LFM on both data sets. For example, in the Yelp data set, the GTRT method improves the LFM method by 12.3% in terms of RMSE. Second, the results for Yelp are generally better than the results for Amazon. This is probably because the

Table 1. Top score words from several topics. *Nouns* are more informative for identifying latent aspects.

Words	Topic	Top Score Terms
Noun	♯6	symbol, teacher, ad, home-study, pollard, singer, card, hardness, development, rudder, notes, technician
	♯12	amazon, system, presentation, bluetooth, piano, update,brand, worktext, head-band, theme, challenge
	♯45	mpvideo, verde, telemarket, tunes, juzz, surprise, year, picture, player, album
Noun, Adj, Adv, Verb	♯11	winter, discuss, report, return, delay, sleep, run, life, painstaking, win
	♯18	disagree, beginner, outstanding, fadeup, soon, highlight, strong, menu, trigger, rudder suitable
	♯27	sound, refer, performance, clock, purchase, divide, control, fadein, laugh, speak, internet

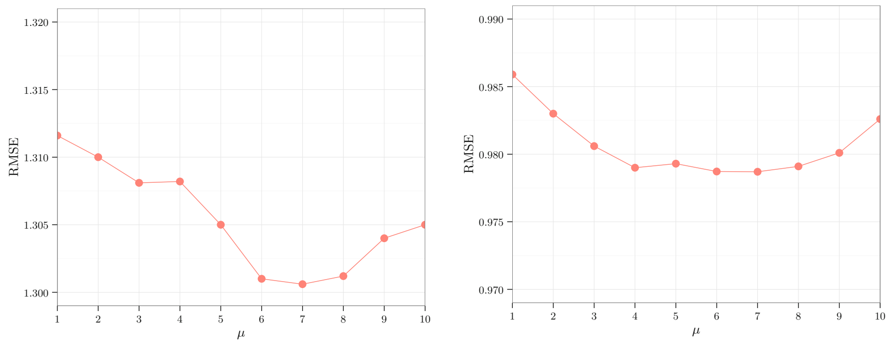
Table 2. Effectiveness comparisons on both data sets. The proposed GTRT method outperforms all the compared methods.

Dataset	Test	MEAN	LFM	GTM	RTM	GTRT
Amazon	10%	1.3423	1.3351	1.3006	1.3218	1.2926
	20%	1.3603	1.3460	1.3111	1.3285	1.3026
Yelp	10%	1.1176	1.1055	0.9787	1.0663	0.9695
	20%	1.1566	1.1489	1.0875	1.1269	1.0762

length of the review content in Yelp is longer than that in Amazon. Overall, this result indicates that the review content can help to improve the prediction of user-item ratings, and the proposed GTRT method effectively leverages the review content.

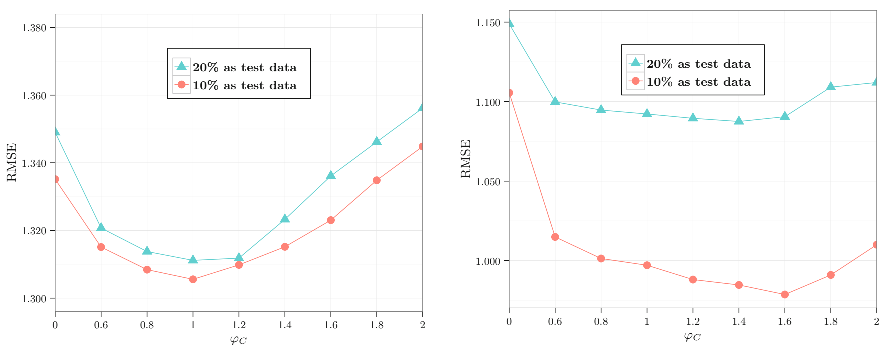
Impact of Parameters φ_C and μ. There are two important parameters in our model, i.e., φ_C and μ. The parameter φ_C is used to control the weight of the review content (i.e., Y_u and Z_i). For example, if we set $\varphi_C = 0$, the GTM method is degenerated to LFM. As we increase φ_C, review content becomes more dominant in the model. Figure 2 shows the parameter study of φ_C. We can see that the choice of φ_C can effect the prediction accuracy and the best choice of φ_C is achieved around 1 - 1.6. We set $\varphi_C = 1.0$ for Amazon and $\varphi_C = 1.6$ for Yelp.

The second parameter μ adjusts the weight between user's own preference and community's overall assessment. Figure 3 shows the parameter study of μ when using 10% of the data as test set. As we can see, the best result is obtained when μ is around 7 in both Amazon and Yelp. In our experiments, we set $\mu = 7$.



(a) Amazon (b) Yelp

Fig. 2. Impact of parameter φ_C



(a) Amazon (b) Yelp

Fig. 3. Impact of parameter μ

Table 3. Result Comparisons for New Users(RMSE). f is the number of topics.

Dataset	$f = 20$		$f = 50$	
	LFM	GTRT	LFM	GTRT
Amazon	1.4302	1.3669	1.4199	1.3529
Yelp	1.0676	1.0158	1.1698	0.9948

Prediction for Cold Start Users. Finally, we test the ability of the proposed method under the cold-start scenario. Intuitively, review content provides additional information that might be valuable for cold start users. In our experiments, we consider users who have expressed less than 5 ratings as cold start users. In both the Amazon data and the Yelp data, more than 50% users are cold start users according to our definition. Table 3 shows the results of LFM and GTRT for the predictions of cold start users. When the factor number is set to 50, GTRT improves the LFM method by 4.7% in Amazon and 14.9% in Yelp. This improvement is greater than the average improvement over all users.

5 Conclusions

In this paper, we have proposed a novel method to explore the review content for better recommendation. Our method first processes the review content along users and items, and then applies the LDA model on nouns only to identify meaningful preference aspects. In particular, two strategies are employed to leverage the review content. The first strategy incorporates review content as a guidance term to guide the learnt latent factors of user preferences; the second strategy formulates a regularization term to constrain the preference differences between similar users. We conduct experimental evaluations on two real data sets, and demonstrate the usefulness of review content as well as the effectiveness of the proposed method for recommendation.

Acknowledgments. This work is supported by the National Basic Research Program of China (No. 2015CB352202), and the National Natural Science Foundation of China (No. 91318301, 61321491, 61100037).

References

1. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. 3, 993–1022 (2003)
2. Lü, L., Medo, M., Yeung, C.H., Zhang, Y.C., Zhang, Z.K., Zhou, T.: Recommender systems. Physics Reports 519, 1–49 (2012)
3. Xue, G.R., Lin, C., Yang, Q., Xi, W., Zeng, H.J., Yu, Y., Chen, Z.: Scalable collaborative filtering using cluster-based smoothing. In: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2005, pp. 114–121. ACM, New York (2005)
4. Khabbaz, M., Lakshmanan, L.V.S.: Toprecs: Top-k algorithms for item-based collaborative filtering. In: Proceedings of the 14th International Conference on Extending Database Technology, EDBT/ICDT 2011, pp. 213–224. ACM, New York (2011)
5. Hofmann, T.: Collaborative filtering via gaussian probabilistic latent semantic analysis. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval, SIGIR 2003, pp. 259–266. ACM, New York (2003)
6. Zhang, Y., Koren, J.: Efficient bayesian hierarchical user modeling for recommendation system. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2007, pp. 47–54. ACM, New York (2007)
7. Salakhutdinov, R., Mnih, A.: Probabilistic matrix factorization. In: Advances in Neural Information Processing Systems, vol. 20. MIT Press, Cambridge (2007)

8. Salakhutdinov, R., Mnih, A.: Bayesian probabilistic matrix factorization using markov chain monte carlo. In: Proceedings of the 25th International Conference on Machine Learning, ICML 2008, pp. 880–887. ACM, New York (2008)
9. Moghaddam, S., Ester, M.: Opinion digger: An unsupervised opinion miner from unstructured product reviews. In: Proceedings of the 19th ACM International Conference on Information and Knowledge Management, CIKM 2010, pp. 1825–1828. ACM, New York (2010)
10. Moghaddam, S., Ester, M.: Ilda: Interdependent lda model for learning latent aspects and their ratings from online product reviews. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2011, pp. 665–674. ACM, New York (2011)
11. Wang, H., Lu, Y., Zhai, C.: Latent aspect rating analysis on review text data: A rating regression approach. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2010, pp. 783–792. ACM, New York (2010)
12. Wang, H., Lu, Y., Zhai, C.: Latent aspect rating analysis without aspect keyword supervision. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2011, pp. 618–626. ACM, New York (2011)
13. Jo, Y., Oh, A.H.: Aspect and sentiment unification model for online review analysis. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, WSDM 2011, pp. 815–824. ACM, New York (2011)
14. Brody, S., Elhadad, N.: An unsupervised aspect-sentiment model for online reviews. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, HLT 2010, pp. 804–812. Association for Computational Linguistics, Stroudsburg (2010)
15. Agarwal, D., Chen, B.C.: flda: Matrix factorization through latent dirichlet allocation. In: Proceedings of the Third ACM International Conference on Web Search and Data Mining, WSDM 2010, pp. 91–100. ACM, New York (2010)
16. McAuley, J., Leskovec, J.: Hidden factors and hidden topics: Understanding rating dimensions with review text. In: Proceedings of the 7th ACM Conference on Recommender Systems, RecSys 2013, pp. 165–172. ACM, New York (2013)
17. Ma, H., Zhou, D., Liu, C., Lyu, M.R., King, I.: Recommender systems with social regularization. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, WSDM 2011, pp. 287–296. ACM, New York (2011)
18. Jindal, N., Liu, B.: Opinion spam and analysis. In: Proceedings of the 2008 International Conference on Web Search and Data Mining, WSDM 2008, pp. 219–230. ACM, New York (2008)