

Research Article

Combining Review Text Content and Reviewer-Item Rating Matrix to Predict Review Rating

Bingkun Wang, Yongfeng Huang, and Xing Li

Tsinghua National Laboratory for Information Science and Technology, Department of Electronic Engineering, Tsinghua University, Beijing 100084, China

Correspondence should be addressed to Bingkun Wang; bkwang77@163.com

Received 22 July 2015; Revised 12 October 2015; Accepted 13 October 2015

Academic Editor: Cheng-Jian Lin

Copyright © 2016 Bingkun Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

E-commerce develops rapidly. Learning and taking good advantage of the myriad reviews from online customers has become crucial to the success in this game, which calls for increasingly more accuracy in sentiment classification of these reviews. Therefore the finer-grained review rating prediction is preferred over the rough binary sentiment classification. There are mainly two types of method in current review rating prediction. One includes methods based on review text content which focus almost exclusively on textual content and seldom relate to those reviewers and items remarked in other relevant reviews. The other one contains methods based on collaborative filtering which extract information from previous records in the reviewer-item rating matrix, however, ignoring review textual content. Here we proposed a framework for review rating prediction which shows the effective combination of the two. Then we further proposed three specific methods under this framework. Experiments on two movie review datasets demonstrate that our review rating prediction framework has better performance than those previous methods.

1. Introduction

Web 2.0 and e-commerce give rise to the explosion of online reviews. In turn, intelligently learning the sentiment propensity and opinions of these reviews is exactly the key to success in current wave of e-commerce. Binary classification, or positive-negative classification, of these reviews has been quite common but it gradually fails to meet the requirement of accuracy [1]. For instance, which item will be selected out of several items that all belong to the positive category is, therefore, hard to predict. However, even the nuances between review rates can lead to great difference in their volume of sales. The review rating prediction research of [2] shows that consumers are often willing to pay 20% to even 99% more money to buy a product of 5-star rate than that of 4-star rate.

There are mainly two types of methods of current review rating prediction (RRP). The first one based on review text content adopts the perspective of natural language process. Researchers transform review text into feature vectors and then employ a multiclass classifier or a regression model to predict review rates [3–7]. It simply ignores the relationship

between the costumers and the items. The second one based on collaborative filtering (CF) focuses on the standpoints of recommenders [8, 9]. Researchers employ the k -nearest neighbor methods [10, 11] or matrix factorization methods [12–17] to extract information from previous reviewer-item rating matrix for review rating prediction. This type of method exploits no information from review text.

In order to include more information to accomplish finer-grained review rating prediction, we proposed a framework combining review text content with previous reviewer-item rating matrix. After that, we contrived three specific methods under this framework. Then we did some experiments on two movie review datasets to examine the efficiency of our framework and three methods. And the result shows that our methods under this framework by and large refine the performance of RRP and generate desirable results.

The outline of the paper is as follows. Section 2 introduces those related researches on RRP. RRP based on review text content and RRP based on CF are described, respectively, in Section 3. Section 4 presents our framework and the three methods under it. Experimental results on two movie review datasets are reported in Section 5. Finally, Section 6

concludes the paper and points out our future research direction.

2. Related Work

The research of sentiment classification is mainly divided into two aspects [18, 19]: one is positive-negative binary classification; the other is the fine-grained RRP. As to binary classification of reviews, some of the creative methods for achieving this have been discussed in [20, 21]. In RRP, there are many research results proposed [3–17], but accuracy of RRP still cannot meet the real demands at present.

At present, there are mainly two ways of fulfilling finer-grained RRP. One includes methods based on review text content (MBRTC), which mines information from review text content by discerning and quantifying a variety of text features and then employs the regression model to predict review rating [7, 22]. For example, Qu et al. [7] consider RRP as a feature engineering problem, and extracted various features, such as words, patterns, syntactic structures, and semantic topics, from the review text to improve the performance. Wang et al. [22] proposed a type of methods based on the content of review and weighting strong social relation of reviewers to predict review rating. To be specific, they predict review rating by incorporating the character of reviewer’s social relations, as regularization constraints, into content-based methods. The main problem of MBRTC is that it mainly uses the review text content information and does not refer to reviewer-items rating matrix information.

There are also some researches based on review text content taking into account characteristics of the items or the reviewers [5, 23]. Wang et al. [23] noticed that the score that a piece of review relates to cannot be fully determined by the review content itself, since review content is not an absolute metric of sentiment propensity. A tough reviewer may use tough words for all items, even items that he rates high. Different items have to meet different basic requirements. Simply analyzing the review content is not enough. Li et al. [5] proposed a method of incorporating reviewer and item information into review text content. They consider the personal characteristics of the reviewers when mining reviews content and use tensor factorization techniques to learn parameter of regression model and predict reviews rating. This method only considers the effect of reviewer and item on review text content and then uses review text content to predict review rating, which is indeed a method based on review text content.

The other one contains methods based on CF (MBCF). Those methods can be further divided into two categories. The first one dictates people to do similarity calculation to find the k -nearest neighbor reviewers or items to do prediction [11, 12]. The second one requires people to use the latent factor model to fulfill the matrix factorization. Several low-dimensional matrix factorization techniques are presented in [24–26]. Koren et al. [12–14] proposed several enhanced matrix factorization methods which can generate promising results by applying heterogeneous information to object functions. Koren [12] built a combined model by merging the matrix factorization and neighborhood models

and improved accuracy of recommendation by extending the models to exploit both explicit and implicit feedback by the users. Koren [14] proposed a method to model the time changing behavior throughout the life span of the data and improved the performance of recommendation. In [27], researchers extended the matrix factorization objective function with the social network information of reviewers. In [28], Shi et al. proposed a context-aware movie recommendation algorithm based on joint matrix factorization (JMF). They jointly factorize the user-item matrix containing general movie ratings and other contextual movie similarity matrixes to integrate contextual information into the recommendation process.

Up to now, some researches combining ratings and text reviews have been applied to recommend system [29, 30]. For example, Cremonesi et al. proposed a hotel recommender algorithm (Interleave), which provides recommendations based on the text reviews and ratings [29]. Levi et al. proposed a recommender system that combines reviews and ratings to recommend hotels [30]. But as far as we know, the methods based on combining ratings and text reviews have not been applied in review rating prediction. Different from the existing methods focusing on recommend system, we focused on review rating prediction and proposed a general framework and three special methods based on review text content and reviewer-item matrix.

Different from the above methods, we propose a framework combining MBRTC and MBCF to include more information to improve the accuracy of prediction. We also present three specific methods under this framework. Finally, the experiment results verify effectiveness of our proposed framework and methods.

3. RRP Based on Review Text Content and RRP Based on Reviewer-Item Rating Matrix

For an online review site with N items $I = \{i_1, i_2, \dots, i_N\}$, T reviews $R = \{r_1, r_2, \dots, r_T\}$ about the N items, M reviewers $A = \{a_1, a_2, \dots, a_M\}$ who have written the T reviews, and $T1$ review ratings $V1 = \{v_1, v_2, \dots, v_{T1}\}$ which are corresponding to $T1$ reviews $R1 = \{r_1, r_2, \dots, r_{T1}\}$, our goal is to predict rating of each review in $R2 = \{r_{T1}, r_{T1+1}, \dots, r_T\}$. In this section, we introduce two existing types of methods in review rating prediction. One is based on review text content; the other one is based on collaborative filtering.

3.1. RRP Based on Review Text Content. Review text content is a very important information source for RRP. Current review text content-based RRP methods mainly use vector space model (VSM) to express review text content and then use linear regression model to predict review rating. To be specific, there are four steps to take. Firstly, online review text content, which includes segmentations of terms, part-of-speech tagging, and frequency statistics, should be pre-processed. Secondly, regarding words, phrases, and n -gram as features, people employ some feature selection methods to choose features that can perfectly express the review text content to compose the feature set $F = \{f_1, f_2, \dots, f_L\}$. Thirdly, each online review r_i in $R = \{r_1, r_2, \dots, r_T\}$ is

expressed as an L -dimensional vector which is exactly an instantiated value of F . Fourthly, the linear regression model dealing with those vectors of reviews is adopted to predict review rating. The linear regression model is described in

$$\hat{v}_i = w^T r_i. \quad (1)$$

To work out the parameter vector w , given training datasets with $R1 = \{r_1, r_2, \dots, r_{T1}\}$ and $V1 = \{v_1, v_2, \dots, v_{T1}\}$ available, least squares error loss is used to minimize the objective function:

$$\min_w \sum_{\text{trainsets}} (v_i - w^T r_i)^2 + \lambda \|w\|^2. \quad (2)$$

Here, $\|w\|^2$, the regularization term of parameter vector w , is employed to avoid overfitting; λ is the regularization coefficient. To estimate the parameters w , a simple stochastic gradient descent algorithm is adopted to solve the optimization problem. For each observed rating $v_i \in V1 = \{v_1, v_2, \dots, v_{T1}\}$, we refer to the following updating rules to learn the parameters w :

$$w \leftarrow w + \eta(\varepsilon_i r_i - \lambda w). \quad (3)$$

Here $\varepsilon_i = v_i - w^T r_i$, and η are the learning rates. After getting w , given $R2 = \{r_{T1}, r_{T1+1}, \dots, r_T\}$, we can apply $\hat{v}_i = w^T r_i$ to predict the review rating of each review in $R2$.

3.2. RRP Based on Collaborative Filtering. RRP now plays an essential role in recommend system. At present, RRP for recommend system mainly based on collaborative filtering involves two methods. One uses the k -nearest neighbor to predict and estimate the current object. The other one uses matrix factorization.

3.2.1. RRP Based on the k -Nearest Neighbor Model. RRP based on the k -nearest neighbor model includes the reviewer-based method and the item-based method. With the reviewer-item rating matrix available, a typical reviewer-based approach is to predict a reviewer's rating on a target item by aggregating the previous ratings on it from k -nearest reviewers. We can consequently formulate the predicted rating on item i from reviewer x as follows:

$$\hat{v}_{xi} = \sum_C s_{xx'} v_{x'i}. \quad (4)$$

Here, C is the set of k nearest neighbor of reviewer x and $s_{xx'}$ represents the similarity between reviewer x and reviewer x' ; $v_{x'i}$ is review rating from reviewer x' on item i .

To get the parameter $s_{xx'}$, given training datasets with $R1 = \{r_1, r_2, \dots, r_{T1}\}$ and $V1 = \{v_1, v_2, \dots, v_{T1}\}$ available, we have to solve the optimization problem, that is, to minimize the square error loss function below:

$$\min_{s_{xx'}} \sum_{\text{trainsets}} \left(v_{xi} - \sum_C s_{xx'} v_{x'i} \right)^2 + \lambda \|s_{xx'}\|^2. \quad (5)$$

Here $\|s_{xx'}\|^2$ is a regularization term of parameter $s_{xx'}$ aimed at avoiding overfitting and λ is regularization coefficient. Then, a simple stochastic gradient descent algorithm is adopted to solve the optimization problem. For each observed rating $v_{xi} \in V1 = \{v_1, v_2, \dots, v_{T1}\}$, we refer to the following updating rules to learn the parameters $s_{xx'}$:

$$s_x \leftarrow s_x + \eta \left(\varepsilon_{xi} \sum_C v_{x'i} - \lambda s_x \right). \quad (6)$$

Here, $\varepsilon_{xi} = v_{xi} - \sum_C s_{xx'} v_{x'i}$ and η are the learning rates. After getting $s_{xx'}$, we can apply $\hat{v}_{xi} = \sum_C s_{xx'} v_{x'i}$ to predict the review rating.

3.2.2. RRP Based on Matrix Factorization. Matrix factorization (MF) is one of the most popular methods in recommend system. The kernel of MF is to find a small number of latent features that might relate to the preferences of reviewer and use them to match observed ratings. A typical model associates each reviewer x with a vector of reviewer factors and each item i with a vector of item factors. The prediction is done through an inner product which is described by

$$\hat{v}_{xi} = p_x q_i^T. \quad (7)$$

In order to compute the two parameters p_x and q_i , we follow the least squares error loss principle to minimize the objective function:

$$\min_{p_x, q_i} \sum_{\text{trainsets}} (v_{xi} - p_x q_i^T)^2 + \lambda (\|p_x\|^2 + \|q_i\|^2). \quad (8)$$

Here $\|p_x\|^2$ and $\|q_i\|^2$ are the regularization terms of parameters p_x and q_i , serving to avoid overfitting; λ is the regularization coefficient. In order to estimate the parameters p_x and q_i , a simple gradient descent algorithm was successfully applied to solve the optimization problem. For each observed rating $v_{xi} \in V1 = \{v_1, v_2, \dots, v_{T1}\}$, we used the following updating rules to acquire the parameters p_x and q_i :

$$\begin{aligned} p_x &\leftarrow p_x + \eta(\varepsilon_{xi} q_i - \lambda p_x), \\ q_i &\leftarrow q_i + \eta(\varepsilon_{xi} p_x - \lambda q_i). \end{aligned} \quad (9)$$

Here, $\varepsilon_{xi} = v_{xi} - p_x q_i^T$ and η are the learning rates. After getting p_x and q_i , we can apply $\hat{v}_{xi} = p_x q_i^T$ to predict review rating.

4. RRP by Combining Review Text Content and Reviewer-Item Rating Matrix

4.1. Problem Description. In order to illustrate the problem we study in this paper, a toy example about reviewers, items, review text content, and review rating is shown in Table 1. From the toy example, we can get three types of information: the user-item rating matrix, review text content with corresponding rating, and review text content without corresponding rating. The problem we study is how to

TABLE 1: A toy example.

Reviewers	Item 1	Item 2	Item 3	Item 4
Reviewer 1	Review text content, 5	Review text content, ?	Review text content, 4	
Reviewer 2		Review text content, 3		Review text content, ?
Reviewer 3	Review text content, 4		Review text content, ?	
Reviewer 4	Review text content, ?	Review text content, 2		Review text content, ?
Reviewer 5	Review text content, ?		Review text content, 5	Review text content, 3

effectively predict missing rating of each review in user-item rating matrix. In this section, we propose a new RRP framework combining reviewer-item rating matrix (RIRM) with review text content (RTC). That is, we want to find a function $f : (\text{RIRM}, \text{RTC}) \rightarrow (\text{rating})$.

4.2. General Framework. There are mainly two types of methods in existing RRP. One includes the methods based on review text content, which can be described as a function $f1 : (\text{RTC}) \rightarrow (\text{rating})$. The other one contains the methods based on collaborative filtering, which can be described as a function $f2 : (\text{RIRM}) \rightarrow (\text{rating})$.

The above two types of methods use either review text content or reviewer-item rating matrix, not having made full use of all the information available. Therefore, we proposed a RRP framework combining review text content and reviewer-item rating matrix. The framework is described in

$$f(\text{RIRM}, \text{RTC}) = (1 - \alpha) f1(\text{RTC}) + \alpha f2(\text{RIRM}). \quad (10)$$

According to the general framework, we proposed three specific special RRP methods. In order to improve the performance of the three special RRP methods, we contrived a way to compute the parameters in the three special RRP methods.

4.3. Three Special RRP Methods

4.3.1. RRP Combining Linear Regression Model and k -Nearest Neighbor. We choose the linear regression model as $f1(\text{RTC})$ and k -nearest neighbor to form $f2(\text{RIRM})$. The special methods are described in

$$\hat{v}_{xi} = (1 - \alpha) wr_{xi} + \alpha \sum_{x'} s_{xx'} v_{x'i}. \quad (11)$$

In order to get optimum parameters α , w , $s_{xx'}$, given training datasets with M reviewers $A = \{a_1, a_2, \dots, a_M\}$ who have written $T1$ review ratings $V1 = \{v_1, v_2, \dots, v_{T1}\}$ which are corresponding to $T1$ reviews $R1 = \{r_1, r_2, \dots, r_{T1}\}$, we follow the least squares error loss principle to minimize the objective function:

$$\begin{aligned} & f(w, s_{xx'}) \\ &= \lambda (\|w\|^2 + \|s_{xx'}\|^2) \\ &+ \min_{w, s_{xx'}} \sum_{\text{trainsets}} \left(v_{xi} - (1 - \alpha) wr_{xi} - \alpha \sum_{x'} s_{xx'} v_{x'i} \right)^2. \end{aligned} \quad (12)$$

Here w and $s_{xx'}$ are regularization terms of those parameters aimed at avoiding overfitting; λ is a regularization coefficient. In order to estimate the parameters α , w , and $s_{xx'}$, we firstly traverse α from 0 to 1. Secondly, for each fixed α in training dataset, we then adopt a simple stochastic gradient descent algorithm to solve the optimization problem. For each observed rating $v_{xi} \in V1 = \{v_1, v_2, \dots, v_{T1}\}$, we refer to the following updating rules:

$$\begin{aligned} w &\leftarrow w + \eta [(1 - \alpha) \varepsilon_{xi} r_{xi} - \lambda w], \\ s_x &\leftarrow s_x + \eta \left(\alpha \varepsilon_{xi} \sum_{x'} v_{x'i} - \lambda s_x \right). \end{aligned} \quad (13)$$

Here $\varepsilon_{xi} = v_{xi} - (1 - \alpha) wr_{xi} - \alpha \sum_{x'} s_{xx'} v_{x'i}$, and η are the learning rates. After getting α , w , and $s_{xx'}$, given $R2 = \{r_{T1}, r_{T1+1}, \dots, r_T\}$, we can apply $\hat{v}_{xi} = (1 - \alpha) wr_{xi} + \alpha \sum_{x'} s_{xx'} v_{x'i}$ to predict review rating of each review in $R2$.

4.3.2. RRP Combining Linear Regression Model and Matrix Factorization. Here, we choose linear regression model as $f1(\text{RTC})$ and Matrix Factorization as $f2(\text{RIRM})$. The special methods are described in

$$\hat{v}_{xi} = (1 - \beta) wr_{xi} + \beta p_x q_i^T. \quad (14)$$

To acquire optimum parameters β , w , p_x , and q_i , given training datasets, we minimize the objective function according to the least square error loss principle:

$$\begin{aligned} & f(w, p_x, q_i) \\ &= \lambda (\|w\|^2 + \|p_x\|^2 + \|q_i\|^2) \\ &+ \min_{w, p_x, q_i} \sum_{\text{trainsets}} \left(v_{xi} - (1 - \beta) wr_{xi} - \beta p_x q_i^T \right)^2. \end{aligned} \quad (15)$$

Here w , p_x , and q_i are regularization terms of those parameters serving to avoid overfitting; α is a regularization coefficient. In order to estimate the parameters β , w , p_x , and q_i , we firstly traverse β from 0 to 1. Secondly, for each fixed β in training dataset, we adopt a simple stochastic gradient descent algorithm to solve the optimization problem. For each observed rating $v_{xi} \in V1 = \{v_1, v_2, \dots, v_{T1}\}$, we refer to the following updating rules:

$$\begin{aligned} w &\leftarrow w + \eta [(1 - \beta) \varepsilon_{xi} r_{xi} - \lambda w], \\ p_x &\leftarrow p_x + \eta (\beta \varepsilon_{xi} q_i - \lambda p_x), \\ q_i &\leftarrow q_i + \eta (\beta \varepsilon_{xi} p_x - \lambda q_i). \end{aligned} \quad (16)$$

Here $\varepsilon_{xi} = v_{xi} - (1 - \beta)wr_{xi} - \beta p_x q_i^T$, and η are the learning rates. After learning β , w , p_x , and q_i , for $R2 = \{r_{T1}, r_{T1+1}, \dots, r_T\}$, we can apply $\hat{v}_{xi} = (1 - \beta)wr_{xi} + \beta p_x q_i^T$ to predict the review ratings of it.

4.3.3. RRP by Combining Linear Regression Model, the k -Nearest Neighbor, and Matrix Factorization. In RRP framework, we choose linear regression model as $f1$ (RTC) and k -nearest neighbor and Matrix Factorization as $f2$ (RIRM). The special methods are described in

$$\hat{v}_{xi} = (1 - \alpha - \beta) wr_{xi} + \alpha \sum_{x'} s_{xx'} v_{x'i} + \beta p_x q_i^T. \quad (17)$$

In order to get optimum parameters α , β , w , $s_{xx'}$, p_x , and q_i , given training datasets available, we follow the least square error loss principle to minimize the objective function:

$$\begin{aligned} f(w, s_{xx'}, p_x, q_i) = & \lambda (\|w\|^2 + \|s_{xx'}\|^2 + \|p_x\|^2 \\ & + \|q_i\|^2) + \min_{w, s_{xx'}, p_x, q_i} \sum_{\text{trainsets}} \left(v_{xi} - (1 - \alpha - \beta) wr_{xi} \right. \\ & \left. - \alpha \sum_{x'} s_{xx'} v_{x'i} - \beta p_x q_i^T \right)^2. \end{aligned} \quad (18)$$

Here w , $s_{xx'}$, p_x , and q_i are regularization terms of those parameters serving to avoid overfitting; λ is a regularization coefficient. In order to estimate the parameters α , β , w , $s_{xx'}$, p_x , and q_i , we firstly traverse α from 0 to 1. Secondly, for each fixed α in training datasets, we traverse β from 0 to 1. Thirdly, for each fixed α and β in training datasets, we adopt a simple stochastic gradient descent algorithm to solve the optimization problem. For each observed rating $v_{xi} \in V1 = \{v_1, v_2, \dots, v_{T1}\}$, we refer to the following updating rules:

$$\begin{aligned} w & \leftarrow w + \eta [(1 - \alpha - \beta) \varepsilon_{xi} r_{xi} - \lambda w], \\ s_x & \leftarrow s_x + \eta \left(\alpha \varepsilon_{xi} \sum_{x'} v_{x'i} - \lambda s_x \right), \\ p_x & \leftarrow p_x + \eta (\beta \varepsilon_{xi} q_i - \lambda p_x), \\ q_i & \leftarrow q_i + \eta (\beta \varepsilon_{xi} p_x - \lambda q_i). \end{aligned} \quad (19)$$

Here $\varepsilon_{xi} = v_{xi} - (1 - \alpha - \beta)wr_{xi} - \alpha \sum_{x'} s_{xx'} v_{x'i} - \beta p_x q_i^T$ and η are the learning rates. After learning α , β , w , $s_{xx'}$, p_x , and q_i , for $R2 = \{r_{T1}, r_{T1+1}, \dots, r_T\}$, we can apply $\hat{v}_{xi} = (1 - \alpha - \beta)wr_{xi} + \alpha \sum_{x'} s_{xx'} v_{x'i} + \beta p_x q_i^T$ to predict the review ratings of it.

5. Evaluations

5.1. Datasets and Experimental Setup. In order to verify the performance of our proposed framework and methods, we performed several experiments on two datasets from the popular review site <http://www.douban.com/>. This website is a reviewer-opinion website where reviewers can read and

TABLE 2: Douban movie reviews datasets.

Datasets	Reviewer	Review	Item	Matrix density
Dataset 1	1476	22593	3041	0.005034
Dataset 2	1079	13858	2087	0.006154

write reviews on movies, music, and books and mark a rating from 1 star to 5 stars. We downloaded information of both movies and reviewers, movie reviews, and their ratings through the API of [http://movie.douban.com/ community](http://movie.douban.com/community). The rough description of the two datasets is shown in Table 2.

In Table 2, reviewer-item rating matrix density (RIRMD) is calculated by

$$\text{RIRMD} = \frac{\text{Number (review)}}{\text{Number (reviewer)} \times \text{Number (item)}}. \quad (20)$$

To evaluate the overall performance of our framework and methods, we divide datasets into 10 parts at random. We do the experiment with taking 80% reviews for training and the remaining 20% reviews for test. We compare our framework and methods with methods based either on review text content or on reviewer-item rating matrix through experiments on two different datasets. The six different methods are abbreviated as follows:

MBLR: methods based on linear regression model.

MBKNN: methods based on k -nearest neighbor.

MBMF: methods based on matrix factorization.

MBLR + MBKNN: our method combining linear regression model and k -nearest neighbor.

MBLR + MBMF: our method combining linear regression model and matrix factorization.

MBLR + MBKNN + MBMF: our method combining linear regression model, k -nearest neighbor, and matrix factorization.

There are mainly two factors influencing RRP in our framework. One is review text content information. The second is review-item matrix information. We did four experiments on two datasets to answer the following four questions:

- (1) How to set parameters α and β , and what are the effects of different parameters α and β on MAE and RMSE of RRP?
- (2) Can our framework and methods decrease MAE and RMSE of RRP?
- (3) Is the algorithm complexity of our methods higher than the three single methods?
- (4) What is the relationship between RIRMD and MAE and RMSE of RRP?

We use root mean square error (RMSE) and mean absolute error (MAE) as metric to evaluate performance of different RRP methods. RMSE and MAE are computed by

$$\begin{aligned} \text{MAE} &= \frac{\sum_{\text{testsets}} |\hat{v}_{xi} - v_{xi}|}{N_{\text{total}}}, \\ \text{RMSE} &= \sqrt{\frac{\sum_{\text{testsets}} (\hat{v}_{xi} - v_{xi})^2}{N_{\text{total}}}}, \end{aligned} \quad (21)$$

where \hat{v}_{xi} is the predicting rating by all kinds of methods, v_{xi} is the rating we have got from test datasets, and N_{total} is the number of reviews in test datasets.

5.2. Performance Evaluation

5.2.1. Setting Parameters α and β in Our Frameworks. In our framework, there are two parameters that have to be set, namely, α and β . We perform 10-fold cross validation in training datasets to get the optimum value of parameters α and β . Figure 1 shows how MAE and RMSE of MBLR + MBKNN change with parameter α in two different training datasets. Figure 2 shows how MAE and RMSE of MBLR + MBMF change with parameter β in two different training datasets. Figure 3 shows how MAE and RMSE of MBLR + MBKNN + MBMF change with parameter β in two different training datasets.

From Figures 1, 2, and 3, we can see that MAE and RMSE of RRP in our methods change with different parameters α and β , which proved that parameters α and β play very important roles in our methods. So, we need to find optimum parameters α and β in our methods to improve the performance of RRP.

In Figure 1, when parameter $\alpha = 0.01$, MAE and RMSE of MBLR + MBKNN are the lowest. The reason is that reviewer-items rating matrix of dataset 1 and dataset 2 is very sparse. It is well known that k -nearest neighbor method based on CF has poor performance on very sparse datasets. At the same time, linear regression model based on review text content has better performance than k -nearest neighbor method based on CF in dataset 1 and dataset 2.

In Figure 2, when parameter β is set between 0.5 and 0.6, MAE and RMSE of MBLR + MBMF are the lowest. The reason is that MF method based on CF has slightly better performance than linear regression model based on review text content.

In Figure 3, we fix $\alpha = 0.01$. When parameter β is chosen between 0.5 and 0.6, MAE and RMSE of MBLR + MBKNN + MBMF are the lowest. The reason is that both MF method based on CF and linear regression model based on review text content have better performance than k -nearest neighbor method based on CF when reviewer-items rating matrix of datasets is very sparse. At the same time, MF method based on CF has slightly better performance than linear regression model based on review text content.

Therefore, we set those parameters of our three methods according to the above results. The special parameters of the three different methods are shown in Table 3.

TABLE 3: Parameters α and β of three methods.

Methods	MBLR + MBKNN	MBLR + MBMF	MBLR + MBKNN + MBMF
Parameter	$\alpha = 0.01$	$\beta = 0.55$	$\alpha = 0.01, \beta = 0.55$

The MAE and RMSE of the three single methods are changed in dataset 1 and dataset 2, but the relative performance of the three single methods is basically unchanged in dataset 1 and dataset 2. For example, the gap between RMSE of the MBKNN and MBLR in dataset 1 and dataset 2 is basically identical (0.353). The gap between RMSE of the MBLR and MBMF in dataset 1 and dataset 2 is basically identical (0.035). So when we combining the three different single methods, we can preliminary choose parameter according to the relative performance of the three different single methods. The gap between RMSE of MBLR and MBMF is 10 times bigger than the gap between RMSE of MBKNN and MBLR.

From Figures 1, 2, and 3, we can see that the performance of our proposed methods is continuously better than single methods when parameter is chosen according to the relative performance of the three different single methods. At the same time, when parameter changed in range of the relative performance of the three different single methods, our proposed methods are continuously better than single methods. For example, we can see that parameter beta changes between 0.1 and 0.9; our methods obtain always better performance than MBKNN, MBLR, and MBMF in Figures 2 and 3. So, in real world applications, our methods, which are obtained by training with one dataset, can be applied to many different ones.

5.2.2. Effect of Different Methods to MAE and RMSE of RRP. In order to verify the performance of our proposed framework and methods, we compare our methods with three baseline methods in two different datasets. The experience results of those six methods on two datasets are presented in Table 4.

In both two datasets, MAE and RMSE of MBLR + MBKNN are lower than that of MBLR and MBKNN; MAE and RMSE of MBLR + MBMF are lower than that of MBLR and MBMF; MAE and RMSE of MBLR + MBKNN + MBMF are lower than that of MBLR, MBKNN, and MBMF. Experimental results prove that combining text content information and review-item matrix information can enhance the performance of RRP. This is because both of the two different information sources are not redundant and can therefore play their own role. When we combine the two types of information: text content information and review-item matrix information, performance of RRP is improved in a certain extent.

For example, MBLR + MBKNN decreases 4.7% ((1.100798 - 1.049105)/1.100798) RMSE compared to MBLR in dataset 1. MBLR + MBMF decreases 5.73% ((1.065305 - 1.004237)/1.065305) RMSE compared to MBMF in dataset 1.

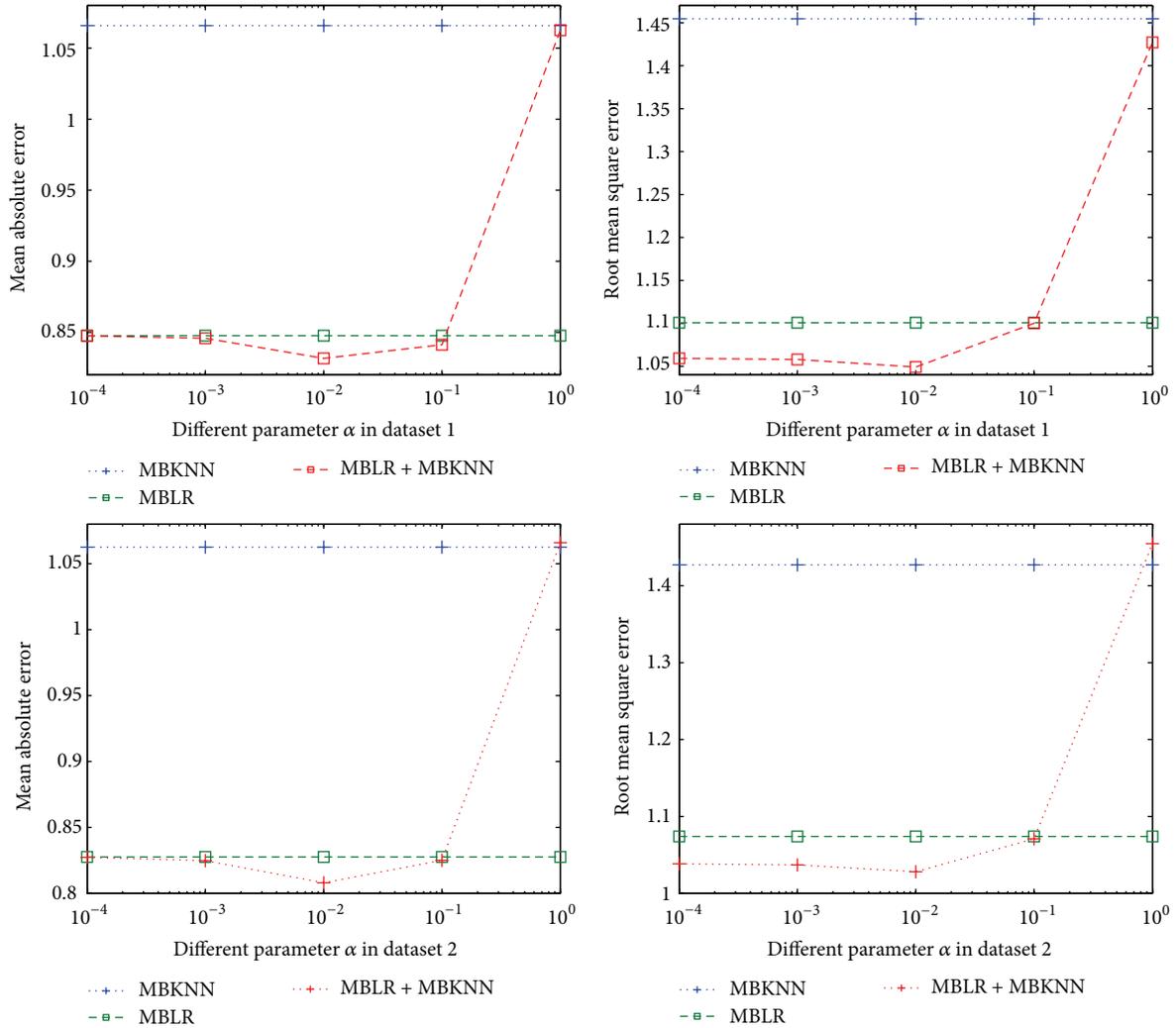
FIGURE 1: MAE and RMSE of MBLR + MBKNN changes with parameter α .

TABLE 4: Douban movie reviews datasets.

Datasets	Metrics	MBKNN	MBLR	MBMF	MBLR + MBKNN	MBLR + MBMF	MBLR + MBKNN + MBMF
Dataset 1	MAE	1.065911	0.847674	0.834128	0.831585	0.802522	0.789961
Dataset 1	RMSE	1.454650	1.100798	1.065305	1.049105	1.004237	0.995946
Dataset 1	Complexity	$O(k)$	$O(L)$	$O(2S)$	$O(L + k)$	$O(L + 2S)$	$O(L + k + 2S)$
Dataset 2	MAE	1.062634	0.827660	0.805563	0.808076	0.747034	0.740456
Dataset 2	RMSE	1.427109	1.074086	1.038729	1.028210	0.971285	0.969373
Dataset 2	Complexity	$O(k)$	$O(L)$	$O(2S)$	$O(L + k)$	$O(L + 2S)$	$O(L + k + 2S)$

L : dimension of VSM; k : number of reviewer's nearest neighbor; S : number of latent factor in ratings matrix.

MBLR + MBKNN + MBMF decreases 6.51% ($(1.065305 - 0.995946)/1.065305$) RMSE compared to MBMF in dataset 1. Although MBLR + MBKNN + MBMF only decreases 0.78% RMSE compared to MBLR + MBMF, both the two methods are our proposed methods. Compared to the three single methods, our proposed methods all decrease the RMSE of RRP and improve the performance of RRP.

5.2.3. Analyzing Complexity of Different Methods. Firstly, we compute the algorithm complexity of three simple

approaches in Section 3. When computing the parameter of the objective function in the three methods, a simple stochastic gradient descent algorithm is adopted. According to the complexity of the stochastic gradient algorithm, we can get the algorithm complexity of the three simple approaches in Section 3. The algorithm complexity of those three methods on two datasets is presented in Table 4.

Then we compute the algorithm complexity of three special RRP approaches in Section 4. Similar to Section 3, when computing the parameter of the three special RRP

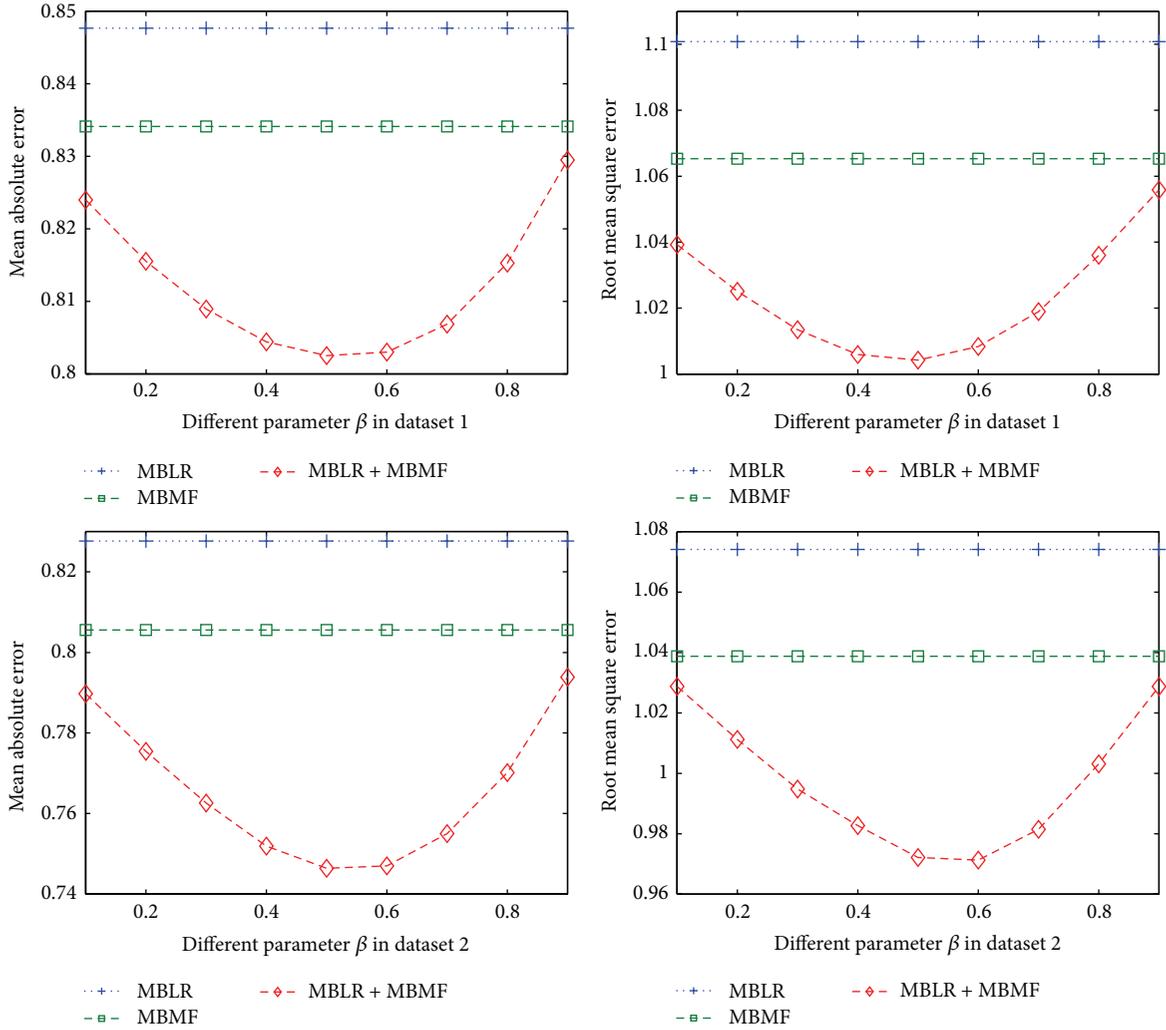


FIGURE 2: MAE and RMSE of MBLR + MBMF changes with parameter β .

methods, we also adopt a simple stochastic gradient descent algorithm to solve the optimization problem of minimizing the objective function. According to the complexity of the stochastic gradient descent algorithm and the computing course of the three special RRP methods, we can get the algorithm complexity of the three specific special RRP methods. The algorithm complexity of those three methods on two datasets is presented in Table 4.

From Table 4, we can see that the algorithm complexity of our proposed three methods in Section 4 is one order of magnitude with the algorithm complexity of the three simple methods in Section 3. When our methods obtain better results than the individual methods, the cost of our method is acceptable.

5.2.4. The Relations between Performance of RRP and RIRMD.

In order to evaluate the effect that the reviewer-item rating matrix density may have on review rating prediction, we experimented on two movie review datasets which are different in RIRMD. The experimental results are shown in Figures 4 and 5.

RIRMD of dataset 2 is denser than dataset 1. From Figures 4 and 5, we can see that MAE and RMSE values of dataset 2 are always lower than that of dataset 1 in our methods, which suggests that higher RIRMD always brings about lower MAE and RMSE. This is because higher RIRMD means more sufficient reviewer-item matrix information. When we combine more sufficient reviewer-item matrix information with review text content, we can get more accurate result statistically.

6. Conclusion

In this paper, we studied the previous methods of RRP and proposed a RRP framework combining review text content and reviewer-item rating matrix to make full use of all information sources to improve the performance of prediction. Based on RRP framework, we further contrived three specific RRP methods. Our methods have significantly enhanced the performance of RRP, compared to methods based solely on review text content or collaborative filtering. In the future, we will further experiment on frameworks

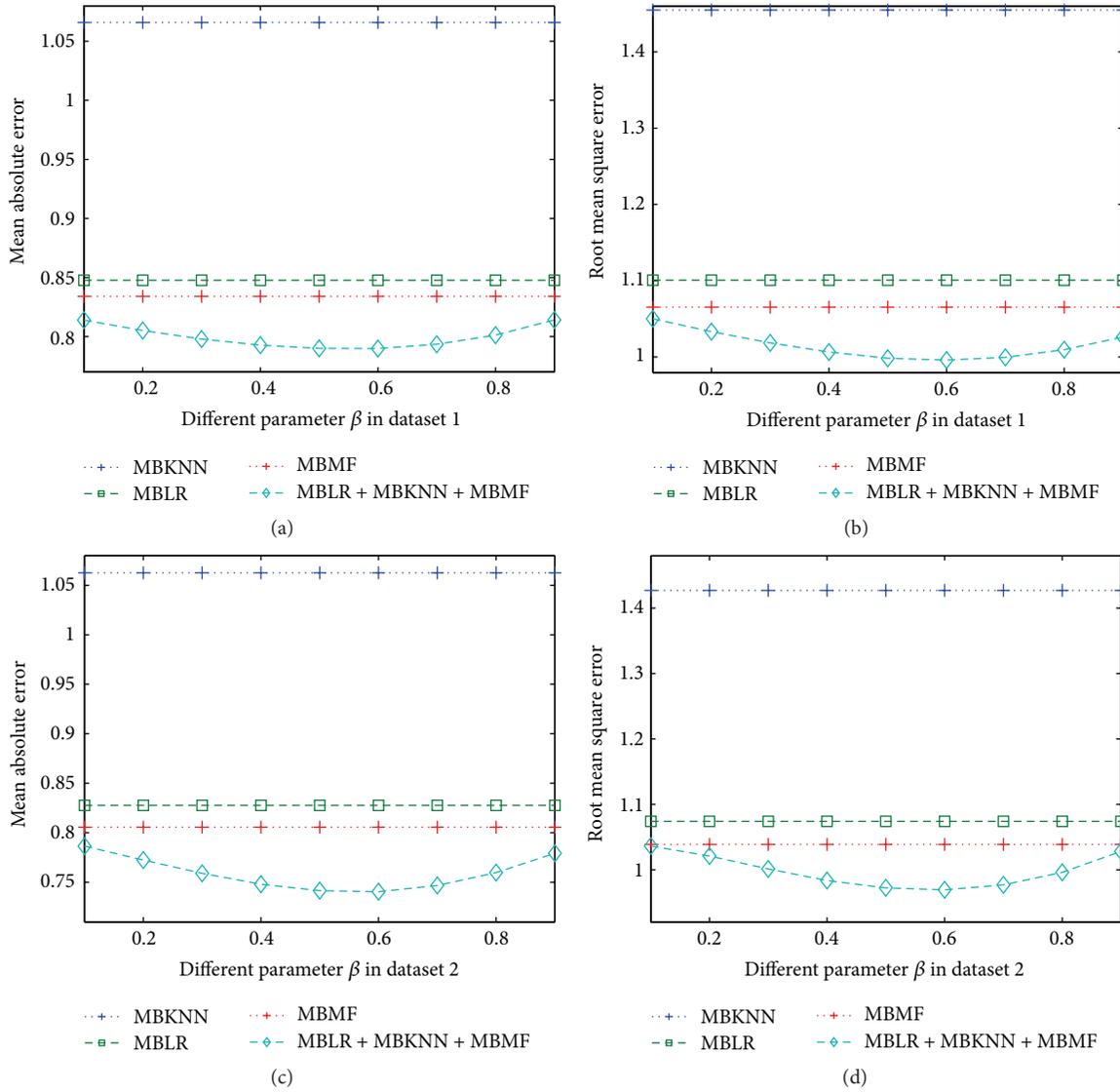


FIGURE 3: MAE and RMSE of MBLR + MBKNN + MBMF changes with parameter β .

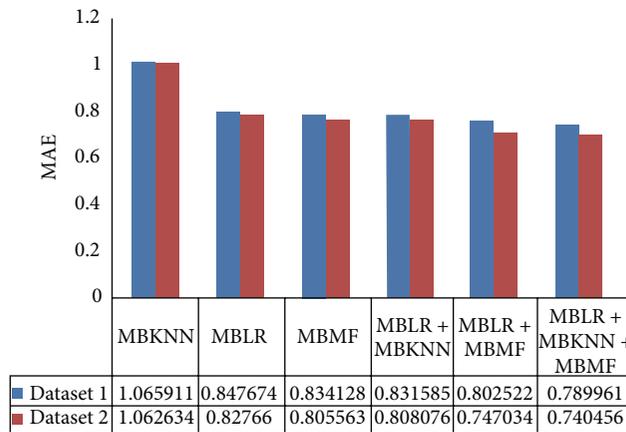


FIGURE 4: Relations between MAE of RRP and RIRMD.

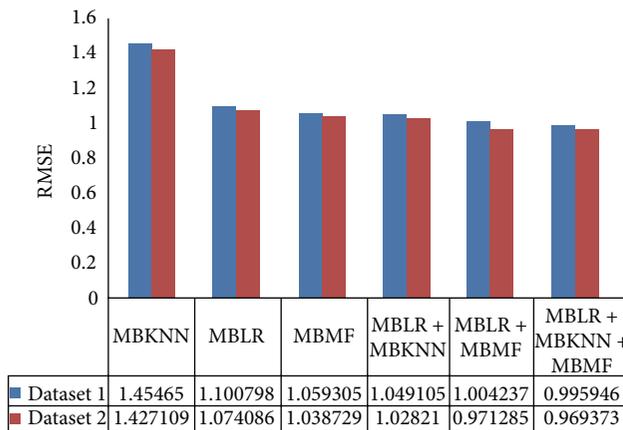


FIGURE 5: Relations between RMSE of RRP and RIRMD.

combining review text content and reviewer-item matrix while employing the probability graph models.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

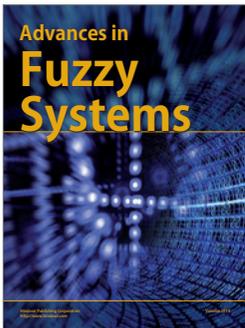
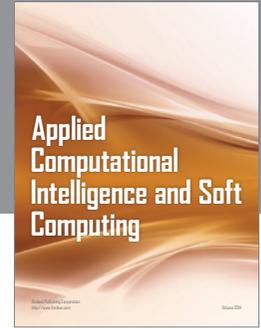
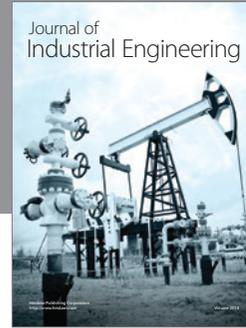
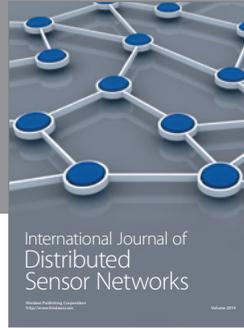
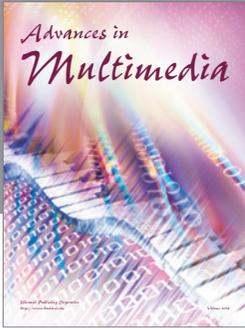
Acknowledgments

The authors thank Lei Liu and Yuchong Li for helpful comments. This work was supported in part by National Key Technology Support Program (2014BAH41B00) and National Natural Science Foundation of China (U1405254, 61472092, 61402115, and 61271392).

References

- [1] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up?: sentiment classification using machine learning techniques," in *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing—Volume 10 (EMNLP '02)*, pp. 79–86, Association for Computational Linguistics, Stroudsburg, Pa, USA, July 2002.
- [2] J. Horrigan, "Online shopping," Pew Internet and American Life Project Report, Pew Research Center, 2008.
- [3] B. Pang and L. Lee, "Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL '05)*, pp. 115–124, June 2005.
- [4] S. Moghaddam and M. Ester, "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 '11)*, pp. 665–674, ACM, Beijing, China, July 2011.
- [5] F. Li, N. Liu, H. Jin, K. Zhao, Q. Yang, and X. Zhu, "Incorporating reviewer and product information for review rating prediction," in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI '11)*, vol. 3, pp. 1820–1825, AAAI Press, Barcelona, Spain, July 2011.
- [6] G. Ganu, N. Elhadad, and A. Marian, "Beyond the stars: improving rating predictions using review text content," in *Proceedings of the 12th International Workshop on the Web and Databases (WebDB '09)*, pp. 1–6, 2009.
- [7] L. Qu, G. Ifrim, and G. Weikum, "The bag-of-opinions method for review rating prediction from sparse text patterns," in *Proceedings of the 23rd International Conference on Computational Linguistics (COLING '10)*, pp. 913–921, Association for Computational Linguistics, Beijing, China, August 2010.
- [8] Y. Shi, M. Larson, and A. Hanjalic, "Collaborative filtering beyond the user-item matrix: a survey of the state of the art and future challenges," *ACM Computing Surveys*, vol. 47, no. 1, Article ID 2556270, pp. 3:1–3:45, 2014.
- [9] H. Ma, "An experimental study on implicit social recommendation," in *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '13)*, pp. 73–82, ACM, New York, NY, USA, August 2013.
- [10] M. Deshpande and G. Karypis, "Item-based top-n recommendation algorithms," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 143–177, 2004.
- [11] R. Jin, J. Y. Chai, and L. Si, "An automatic weighting scheme for collaborative filtering," in *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '04)*, pp. 337–344, ACM, Sheffield, UK, July 2004.
- [12] Y. 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 (KDD '08)*, pp. 426–434, August 2008.
- [13] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.
- [14] Y. Koren, "Collaborative filtering with temporal dynamics," *Communications of the ACM*, vol. 53, no. 4, pp. 89–97, 2010.
- [15] M. Kurucz, A. A. Benczúr, and B. Torma, "Methods for large scale svd with missing values," in *Proceedings of the KDD Cup and Workshop*, San Jose, Calif, USA, August 2007.
- [16] R. Salakhutdinov and A. Mnih, "Bayesian probabilistic matrix factorization using markov chain Monte Carlo," in *Proceedings of the 25th International Conference on Machine Learning (ICML '08)*, pp. 880–887, ACM, Helsinki, Finland, July 2008.
- [17] K. Yu, S. Zhu, J. Lafferty, and Y. Gong, "Fast nonparametric matrix factorization for large-scale collaborative filtering," in *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '09)*, pp. 211–218, ACM, Boston, Mass, USA, July 2009.
- [18] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Foundations and Trends in Information Retrieval*, vol. 2, no. 1-2, pp. 1–135, 2008.
- [19] B. Liu, *Sentiment Analysis and Opinion Mining*, Synthesis Lectures on Human Language Technologies, Morgan & Claypool Publishers, 2012.
- [20] B. Wang, Y. Min, Y. Huang et al., "Chinese reviews sentiment classification based on quantified sentiment lexicon and fuzzy set," in *Proceedings of the 3rd International Conference on Information Science and Technology (ICIST '13)*, pp. 678–680, IEEE, Yangzhou, China, March 2013.
- [21] A. Bickerstaffe and I. Zukerman, "A hierarchical classifier applied to multi-way sentiment detection," in *Proceedings of*

- the 23rd International Conference on Computational Linguistics (COLING '10)*, pp. 62–70, Association for Computational Linguistics, Beijing, China, August 2010.
- [22] B. Wang, Y. Min, Y. Huang, X. Li, and F. Wu, “Review rating prediction based on the content and weighting strong social relation of reviewers,” in *Proceedings of the International Workshop on Mining Unstructured Big Data Using Natural Language Processing (UnstructureNLP '13)*, pp. 23–30, ACM, Burlingame, Calif, USA, October–November 2013.
- [23] H. Wang, Y. Lu, and C. Zhai, “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 '10)*, pp. 783–792, ACM, Washington, DC, USA, July 2010.
- [24] J. D. M. Rennie and N. Srebro, “Fast maximum margin matrix factorization for collaborative prediction,” in *Proceedings of the 22nd International Conference on Machine Learning (ICML '05)*, pp. 713–719, August 2005.
- [25] A. Mnih and R. Salakhutdinov, “Probabilistic matrix factorization,” in *Proceedings of the Advances in Neural Information Processing Systems (NIPS '07)*, pp. 1257–1264, Vancouver, Canada, 2007.
- [26] N. S. Nati and T. Jaakkola, “Weighted low-rank approximations,” in *Proceedings of the 20th International Conference on Machine Learning*, pp. 720–727, AAAI Press, Washington, DC, USA, August 2003.
- [27] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, “Recommender systems with social regularization,” in *Proceedings of the 4th ACM International Conference on Web Search and Data Mining (WSDM '11)*, pp. 287–296, ACM, Hong Kong, February 2011.
- [28] Y. Shi, M. Larson, and A. Hanjalic, “Mining contextual movie similarity with matrix factorization for context-aware recommendation,” *ACM Transactions on Intelligent Systems and Technology*, vol. 4, no. 1, article 16, 2013.
- [29] P. Cremonesi, F. Garzotto, and R. Turrin, “User-centric vs. system-centric evaluation of recommender systems,” in *Human-Computer Interaction—INTERACT 2013*, P. Kotz, G. Marsden, G. Lindgaard, J. Wesson, and M. Winckler, Eds., vol. 8119 of *Lecture Notes in Computer Science*, pp. 334–351, Springer, Berlin, Germany, 2013.
- [30] A. Levi, O. Mokryn, C. Diot, and N. Taft, “Finding a needle in a haystack of reviews: cold start context-based hotel recommender system demo,” in *Proceedings of the 6th ACM Conference on Recommender Systems (RecSys '12)*, pp. 305–306, ACM, Dublin, Ireland, September 2012.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

