information

MDPI

Article

User-Personalized Review Rating Prediction Method Based on Review Text Content and User-Item Rating Matrix

Bingkun Wang [1], Bing Chen [2], Li Ma [1,*] and Gaiyun Zhou [1]

[1] School of Computer, Pingdingshan University, Pingdingshan 467000, China; bkwang77@pdsu.edu.cn (B.W.); zhoudeligent@163.com (G.Z.)
[2] Huanghe Science & Technology University, Zhengzhou 450000, China; 13373900266@189.cn
* Correspondence: malixnc@126.com; Tel.: +86-151-3752-9916

Received: 9 November 2018; Accepted: 18 December 2018; Published: 20 December 2018

Abstract: With the explosive growth of product reviews, review rating prediction has become an important research topic which has a wide range of applications. The existing review rating prediction methods use a unified model to perform rating prediction on reviews published by different users, ignoring the differences of users within these reviews. Constructing a separate personalized model for each user to capture the user's personalized sentiment expression is an effective attempt to improve the performance of the review rating prediction. The user-personalized sentiment information can be obtained not only by the review text but also by the user-item rating matrix. Therefore, we propose a user-personalized review rating prediction method by integrating the review text and user-item rating matrix information. In our approach, each user has a personalized review rating prediction model, which is decomposed into two components, one part is based on review text and the other is based on user-item rating matrix. Through extensive experiments on Yelp and Douban datasets, we validate that our methods can significantly outperform the state-of-the-art methods.

Keywords: review rating prediction; sentiment classification; user-item matrix; user-personalized model

1. Introduction

Web 2.0 and e-commerce have triggered an explosion of online reviews. These reviews usually contain a large amount of sentiment and opinion information that is essential to many decision-making processes, such as personalized consumption decisions, product quality tracking, and public opinion mining. How to mine the information of reviews on sentiment and opinions has become a fundamental problem in natural language processing (NLP) and Web mining fields [1,2].

Sentiment polarity classification of online reviews has been widely studied in NLP, but it gradually fails to meet the requirement for mining fine-grained sentiment [3–7]. For example, a consumer doesn't know how to choose the optimum product from all kinds of products when they all belong to the positive sentiment polarity. Some studies have shown that consumers are willing to pay 20% to 99% extra for five-star ratings rather than four-star ratings [8]. This indicates that slight differences in product ratings may lead to dramatic changes in product sales. For opinion mining, the government should not only understand the positive and negative sentiment polarities but also further understand the intensity of positive and negative sentiments in order to distinguish the urgency of public opinion events and take different measures. Therefore, researchers are increasingly concerned with review rating predictions (RRP). Existing RRP methods based on the review text content mainly transform review text into feature vectors and then employ a machine learning model to predict review rates [9–13]. For example, RRP is considered as a feature engineering problem, and the performance of

RRP is improved by extracting different features, such as words, lexical patterns, syntactic structures, and semantic topics from the review text content [10]. Zhang et al. extracted the feature from review text content through word embedding and a Convolutional Neural Network CNN and then realized the RRP through the fully connected network. In this way, the performance of the RRP is improved [13].

The RRP methods based on review text content have an implicit assumption that the sentiment magnitude expressed by different users using the same sentiment words is consistent, and the sentiment magnitude expressed by different sentiment words is different. However, this implicit hypothesis does not match the actual situation. For example, different users providing similar reviews on a product might rate it differently, or they might give it the same rating while writing very different reviews, depending on how strict/lenient they are or how they like to convey their opinions. Wang et al. believe that the rating is not entirely determined by the review text content, because a harsh user may comment on all products with strict words, even if they give the product a high rating [14]. Different consumers make use of the same sentiment words to express different sentiment intensities, which reflects the consumer's personalized expression when using sentiment words. Based on the above analysis, we found that the RRP is not only related to the review text content but is also related to the personalized information of the reviewer.

Review text content is an important source of information for obtaining personalized information regarding users. Wu et al. considered the personalized information of micro-blog users, proposed a personalized micro-blog sentiment classification method, and achieved better sentiment classification performance [15]. The user-item rating matrix is another data source for obtaining personalized information about users. From the perspective of the recommendation system, based on the historical rating in the user-item rating matrix, the personalized information of the users can be mined through the collaborative filtering algorithm [16–21].

The main problem with the existing RRP methods based on the review text content is that the user personalization dependency of the sentiment word cannot be fully exploited only based on the review text content. The user personalized information can be obtained not only by the review text content but also by the user-item rating matrix [22]. Therefore, we propose a user-personalized review rating prediction (UPRRP) method based on review text content and user-item rating matrix by integrating the review text content and user-item rating matrix information. Our method firstly models the commonality and personality of the user's sentiment expression based on the review text content and then models user personalization through the user-item rating matrix. Finally, the UPRRP is realized by linearly integrating the review text content and the user-item rating matrix information.

The main contributions of this paper can be summarized as:

(1) We propose a novel method based on review text and user-item rating matrix for personalized review rating prediction.

(2) We model user personality sentiment information by integrating review text and user-item rating matrix information.

(3) Our comparative results on four datasets show that our model is significantly better than previous approaches on tasks of review rating prediction.

The rest of the paper is as follows. Section 2 introduces related researches on RRP. Section 3 describes the three UPRRP methods we proposed. Experimental results on four review datasets are reported in Section 4. Finally, Section 5 concludes the paper and points out the future research direction.

2. Related Work

2.1. RRP Based on Review Content

The existing RRP is mainly implemented by mining the sentiment information contained in the online review text content [23,24]. RRP is proposed by Pang and Lee [23]. RRP is generally formatted as a regression problem because the ratings have a certain order. Pang and Lee implement RRP using Support Vector Machine (SVM) multi-classifiers and SVM regression models, respectively.

The experimental results in [23] proved that an SVM regression model is superior to an SVM multi-classifier in RRP. The reason is that the score prediction is a continuous value rather than a discrete category, so the classification model is not as effective as the regression model.

Qu et al. proposed a bag-of-opinions review text representation model that is different from the traditional bag-of-words text representation model [10]. Since the role of modifiers and negative words is considered in a bag-of-opinions model, the method in [10] finally achieves better performance than the traditional bag-of-words model in [23].

There are some studies that consider the reviewers and items information based on review text content [11,14]. Wang et al. believe that the rating is not entirely determined by the review text content, because a demanding user may review all products with harsh words, even if he gives a higher rating to products [14]. Consistent with [14], a method of merging users and products into review text content is proposed in [11]. Li et al. implemented RRP using the parameters of the tensor factorization learning regression model [11]. Li et al. achieved RRP and learned the parameters of the regression model by using tensor factorization [11].

2.2. Missing Score Prediction in the User-Item Rating Matrix

One study that is highly correlated with RRP is the missing scores prediction for the user-item rating matrix in the recommendation system. The difference between RRP and the missing score prediction in the user-item rating matrix is that the RRP predicts the score based on the user-published review text information and the missing score prediction predicts the missing scores in the user-item rating matrix based on history ratings written by users. Two different types of rating prediction studies implement rating predictions from different perspectives. Therefore, the prediction of missing scores in the user-item rating matrix is an important reference for RRP.

The missing score-prediction methods in a user-item rating matrix mainly include two types, which are K-nearest neighbor (KNN) and matrix factorization (MF). KNN methods mainly include two types of methods, one is KNN based on a user similarity calculation and the other is KNN based on an item similarity calculation [25,26]. The essence of KNN based on the user similarity calculation method is to calculate the similarity between users based on the user-item rating matrix information and then predict the missing score of the target user based on the history rating of the K users with the highest similarity to the target user. The KNN based on the item similarity calculation method is similar to KNN based on the user similarity calculation method, except that the user is replaced by an item. The essence of MF is to project users and items into a shared latent factor space and then use the latent factor vector of the user and the item to model interactions between users and items [16–21]. Recently, there has been a trend of applying deep learning techniques in the recommendation [27,28]. For example, He et al. generalized matrix factorization and factorization machines to neural collaborative filtering and achieved promising performances [28].

2.3. Review-Based Recommendation

When the user-item rating matrix is sparse, the performance of the missing score prediction in the user-item rating matrix will be significantly reduced. Therefore, some research work considers review text content information to improve the performance of missing score predictions in the user-item rating matrix. The effectiveness of using review text content information in recommendation has been widely discussed and demonstrated in many existing research findings [13,29–31].

By incorporating user review text content information, some research efforts generate latent factors for users and items by integrating topic models into the collaborative filtering framework [32–37]. One of the early studies of using review text content to improve missing score predictions in the user-item rating matrix was presented in [38]. The study found that reviews often include information such as price, service, positive or negative sentiments that can be used for missing score predictions in a user-item rating matrix. A hidden factors model was proposed in [34]. This approach has achieved

significant improvements in RRP compared to models that use only a user-item rating matrix or review text content.

Textual reviews have also been used in deep learning models for recommendation [13,29,39,40]. In DeepCoNN, reviews are first processed by two CNNs to learn representations of users and items, which are then concatenated and passed into a regression layer for rating prediction. A limitation of DeepCoNN is that it uses reviews in the testing phase [13]. The performance of DeepCoNN decreases greatly when reviews are unavailable in the testing phase. To deal with the problem, TransNet [29] extends DeepCoNN by introducing an additional layer to simulate the review corresponding to the target user-item pair. The generated review is then used for rating prediction.

The existing review-based recommendation method predicts the missing score in a user-item rating matrix from the history of review text written by a user and the user-item rating matrix. In our paper, we mainly study the rating prediction of an existing review. Existing review-based recommendation methods provide a common RRP model for all users. In contrast, our approach builds a user-specific review score prediction model for each user.

3. UPRRP Based on Review Text Content and a User-Item Rating Matrix

3.1. Problem Description

For an online review site that contains N items $I = \{i_1, i_2, \ldots, i_N\}$ and M users $U = \{u_1, u_2, \ldots, u_M\}$, the M users have published T reviews $R = \{r_1, r_2, \ldots, r_T\}$ on N items. Among them, there are $T1$ reviews $R1 = \{r_1, r_2, \ldots, r_{T1}\}$ that have corresponding ratings $V1 = \{v_1, v_2, \ldots, v_{T1}\}$, and the remaining ($T$–$T1$) reviews $R2 = \{r_{T1}, r_{T1+1}, \ldots, r_T\}$ have no corresponding ratings.

In order to more clearly describe the problem, we designed a toy example in Table 1. In this table, we can get two types of information; user-item rating matrix (UIRM) information and review text content (RTC) information. Our goal is to predict the reviews rating (RR) by using the existing user-item rating matrix and review text content information. That is, we want to find a function *f: (RTC, UIRM) → (RR)* which can be used to compute the rating of reviews.

Table 1. A dummy example.

	Item 1	Item 2	Item 3
User 1	Review text content, 5	Review text content, 3	
User 2	Review text content, ?		Review text content, 4
User 3	Review text content, ?		Review text content, ?
User 4		Review text content, 2	Review text content, ?
User 5		Review text content, ?	

3.2. UPRRP Method 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 a vector space model (VSM) to express review text content and then use a linear regression model to predict the review rating. Specifically, 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 preprocessed. 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. Thirdly, each online review is expressed as a multi-dimensional vector. Finally, the linear regression model dealing with those vectors of reviews is adopted to predict the review rating.

$$\hat{v}_{ui} = \mathbf{w}^T \mathbf{r}_{ui} \tag{1}$$

Here, $\hat{v}_{ui}$ is the predicted score of user u for item i; $\mathbf{w}$ is the parameters of the function; $\mathbf{r}_{ui}$ is the vector representation of review text content.

Because of the difference of sentiment expression among different users in product review sites, the general RRP model established for all users does not accurately understand the particular sentiment information of each user. It is the most intuitive way to design a personalized RRP method for each user by using the personal review text content posted by each user in product review sites. Nevertheless, in product review sites, the personal review text content posted by a single user is generally very scarce. Therefore, based on the personal review text content information alone, it is very difficult to accurately train a UPRRP model for each user.

Social science research shows that while online users express their sentiments in a personalized way, different users share many of the same sentiment expressions [41]. For example, "poor" and "bad" are often used to express negative emotions between different users. Therefore, taking full advantage of the shared sentiment information between different users can effectively solve the problem of insufficient data of individual users.

Based on the above analysis, a UPRRP model based on the review text content (UPRRP+RTC) is proposed. In order to model the sentiment commonality of different users and sentiment personality of a single user, the UPRRP model is decomposed into two parts, one is public and the other is user-specific. The public part shared by all users is used to describe the sentiment information shared by different users. The model parameters in the public part are trained using all the user data. The user-specific portion that is unique to each user is used to describe the specific sentiment expression for each user. The model parameters in the user-special part are trained using the single user's data.

To be specific, user u has published a review $\mathbf{r}_{\mathrm{ui}}$ on the item i. The UPRRP model based on the review text content is as follows:

$$\hat{v}_{ui} = (\mathbf{w} + \mathbf{w}_u)^T \mathbf{r}_{ui} \tag{2}$$

Here, $\hat{v}_{ui}$ is the predicted rating of user u for item i; $\mathbf{w}$ and $\mathbf{w}_{\mathrm{u}}$ are the public and specific parameters in UPRRP model; $\mathbf{r}_{ui}$ is the vector representation of review text content.

To estimate the parameter vectors $\mathbf{w}$ and $\mathbf{w}_u$, given $R1 = \{r_1, r_2, \ldots, r_{T1}\}$ and $V1 = \{v_1, v_2, \ldots, v_{T1}\}$, we minimize the objective function by applying the least squares error loss principle in the training data set.

$$\min_{\mathbf{w},\mathbf{w}_u} \sum_{trainsets} \left(v_{ui} - (w + w_u)^T r_{ui}\right)^2 + \lambda \left(\|\mathbf{w}\|^2 + \|\mathbf{w}_u\|^2 \right) \tag{3}$$

Here, $\|\mathbf{w}\|$ and $\|\mathbf{w}_u\|$ are the regular terms and λ is the regular coefficient. To calculate the parameter vectors $\mathbf{w}$ and $\mathbf{w}_u$, we solve this optimization function by applying a stochastic gradient descent. Finally, we learn the parameters $\mathbf{w}$ and $\mathbf{w}_u$ by using the following update rules.

$$\mathbf{w} \leftarrow \mathbf{w} + \eta(\varepsilon_{ui}\mathbf{r}_{ui} - \lambda \mathbf{w}) \tag{4}$$

$$\mathbf{w}_u \leftarrow \mathbf{w}_u + \eta(\varepsilon_{ui}\mathbf{r}_{ui} - \lambda \mathbf{w}_u) \tag{5}$$

Here, $\varepsilon_{ui} = v_{ui} - (\mathbf{w} + \mathbf{w}_u)^T \mathbf{r}_{ui}$, η is learning rate. After getting $\mathbf{w}$ and $\mathbf{w}_u$, given $R2 = \{r_{T1+1}, r_{T1+2}, \ldots, r_T\}$, we predict the review rating by using $\hat{v}_{ui} = (\mathbf{w} + \mathbf{w}_u)^T \mathbf{r}_{ui}$.

3.3. UPRRP Based on the User-Item Rating Matrix

In the Recommender Systems (RS), the key to personalized modeling and recommendations for users is to predict the score of the missing rating in UIRM based on the historical ratings in the UIRM. The existing mainstream recommendation method is collaborative filtering (CF), which mainly includes two types of methods; K nearest neighbor method (KNN) based on user similarity or item similarity and matrix factorization (MF) method based on the latent factor model.

KNN-based RRP includes KNN based on user similarity and KNN based on item similarity. The ideas of these two methods are basically the same. Since our goal is to achieve RRP by mining the user's personalized information. Therefore, we adapt the KNN based on user similarity.

RRP based on matrix factorization is the most popular method in RS. The core idea of the algorithm is to first find latent factors related to the user's personalized preferences, and then associate the users with the items through the latent factors. By mining the user's personalized information, the user's rating of the item is finally realized.

The two types of methods based on KNN and MF have different perspectives in implementing RRP. Considering the information complementarity, we propose a UPRRP model based on the user-item rating matrix by integrating KNN and MF algorithms.

$$\hat{v}_{ui} = (1-\beta)\sum_{u\prime\in C} s_{uu\prime} v_{u\prime i} + \beta \mathbf{p}_u \mathbf{q}_i^T \tag{6}$$

Here, β is the parameter that must be estimated, which is used to adjust the proportion of KNN and MF in our method. $\hat{v}_{ui}$ is the predicted rating of user u for item i, C is the set of k nearest neighbors of user u, $s_{uu'}$ is the similarity between the user u and the user u', and $v_{u'i}$ is the rating of the item i by the user u'. We define $\mathbf{s}_u$ as a k-dimensional vector which is composed of $s_{uu'}$, and $\boldsymbol{v}_i$ is a k-dimensional vector which is composed of $v_{u'i}$. $\mathbf{p}_u$ is the latent factor vector of user u, $\mathbf{q}_i$ is the latent factor vector of the item i.

To calculate the parameter β, $\mathbf{s}_u$, $\mathbf{p}_u$, and $\mathbf{q}_i$, given the training data set $R1 = \{r_1, r_2, \ldots, r_{T1}\}$ and $V1 = \{v_1, v_2, \ldots, v_{T1}\}$, we use the least-square error loss in training data as the objective function.

$$\min_{s_{uu'},\mathbf{p}_u,\mathbf{q}_i} \sum_{trainsets} \left(v_{ui} - (1-\beta)\sum_{u\prime\in C} s_{uu\prime} v_{u\prime i} - \beta p_u q_i^T\right)^2 + \lambda(\|\mathbf{s}_u\|^2 + \|\mathbf{p}_u\|^2 + \|\mathbf{q}_i\|^2) \tag{7}$$

Here, λ is the regular coefficient, $\|\mathbf{s}_u\|$, $\|\mathbf{p}_u\|$, and $\|\mathbf{q}_i\|$ are the regular terms of the parameter. To estimate the parameter β, $\mathbf{s}_u$, $\mathbf{p}_u$, and $\mathbf{q}_i$, we first traverse β from 0 to 1 in steps of 0.01, and then solve this optimization problem for each fixed β by applying a stochastic gradient descent algorithm in the training dataset. We learn the parameters $\mathbf{s}_u$, $\mathbf{p}_u$, and $\mathbf{q}_i$ by using the following update rules.

$$\mathbf{s}_u \leftarrow \mathbf{s}_u + \eta((1-\beta)\varepsilon_{ui}\mathbf{v}_i - \lambda \mathbf{s}_u) \tag{8}$$

$$\mathbf{p}_u \leftarrow \mathbf{p}_u + \eta(\beta\varepsilon_{ui}\mathbf{q}_i - \lambda \mathbf{p}_u) \tag{9}$$

$$\mathbf{q}_i \leftarrow \mathbf{q}_i + \eta(\beta\varepsilon_{ui}\mathbf{p}_u - \lambda \mathbf{q}_i) \tag{10}$$

Here, $\varepsilon_{ui} = v_{ui} - (1-\beta)\sum_{u\prime\in C} s_{uu\prime} v_{u\prime i} - \beta \mathbf{p}_u \mathbf{q}_i^T$, η is learning rate. After getting β, $\mathbf{s}_u$, $\mathbf{p}_u$, and $\mathbf{q}_i$, given $R2 = \{r_{T1+1}, r_{T1+2}, \ldots, r_T\}$, we can use $\hat{v}_{ui} = (1-\beta)\sum_{u\prime\in C} s_{uu\prime} v_{u\prime i} + \beta \mathbf{p}_u \mathbf{q}_i^T$ to predict the review rating.

3.4. UPRRP Based on Review Text Content and the User-Item Rating Matrix

There are mainly two types of methods in existing RRP. The first one includes the methods based on review text content, which can be described as a function *f1: (RTC) → (RR)*. It simply ignores the relationship between the reviewers and the items. The other one contains the methods based on collaborative filtering, which can be described as a function *f2: (UIRM) → (RR)*. This type of method exploits no information from review text content. Review text content and the user-item rating matrix are two types of different information sources for obtaining users' personalized sentiment information. Based on Sections 3.2 and 3.3, we propose a UPRRP method based on the review text content and the

user-item rating matrix by integrating the review text content information and the user-item rating matrix information.

$$\hat{v}_{ui} = (1-\alpha)(\mathbf{w}+\mathbf{w}_u)^T\mathbf{r}_{ui} + \alpha\left[(1-\beta)\sum_{u'\in C} s_{uu'}v_{u'i} + \beta\mathbf{p}_u\mathbf{q}_i^T\right] \tag{11}$$

Here, β is the parameter which is estimated in Section 3.3, α is parameter that needs to estimated and is used to adjust the proportion of UPRRP based on review text content and UPRRP based on user-item rating in our method. v_{ui} is the predicted rating of user u for item i; $\mathbf{w}$ and $\mathbf{w}_u$ are the common and specific parameters in the UPRRP model; $\mathbf{r}_{ui}$ is the vector representation of review text content. C is the set of k nearest neighbors of user u, $s_{uu'}$ is the similarity between the user u and the user u', and $v_{u'i}$ is the rating of the item i by the user u'. $\mathbf{p}_u$ is the latent factor vector of user u and $\mathbf{q}_i$ is the latent factor vector of the item i.

In order to get the optimum parameters α, $\mathbf{w}$, $\mathbf{w}_u$, $\mathbf{s}_u$, $\mathbf{p}_u$, and $\mathbf{q}_i$, we use the least-square error loss to minimize the objective function in the training datasets.

$$\begin{aligned}\min_{\mathbf{w},\mathbf{w}_u,s_{uu'},\mathbf{p}_u,\mathbf{q}_i} \sum_{trainsets} \left\{v_{ui} - (1-\alpha)(w+w_u)^T r_{ui} - \alpha\left[(1-\beta)\sum_{u'\in C} s_{uu'}v_{u'i} + \beta p_u q_i^T\right]\right\}^2 \\ +\lambda(\|\mathbf{w}\|^2 + \|\mathbf{w}_u\|^2 + \|\mathbf{s}_u\|^2 + \|\mathbf{p}_u\|^2 + \|\mathbf{q}_i\|^2)\end{aligned} \tag{12}$$

Here, λ is the regular coefficient, $\|\mathbf{w}\|$, $\|\mathbf{w_u}\|$, $\|\mathbf{s}_u\|$, $\|\mathbf{p}_u\|$, and $\|\mathbf{q}_i\|$ are the regular terms of the parameter. To estimate the parameter α, $\mathbf{w}$, $\mathbf{w}_u$, $\mathbf{s}_u$, $\mathbf{p}_u$, and $\mathbf{q}_i$, we first get the optimal parameters β based on Section 3.3, then traverse α from 0 to 1 in steps of 0.01, and finally, use a stochastic gradient descent algorithm to solve this optimization problem for each fixed α in the training dataset. We learn the parameters $\mathbf{w}$, $\mathbf{w}_u$, $\mathbf{s}_u$, $\mathbf{p}_u$, and $\mathbf{q}_i$ by applying the following update rules.

$$\mathbf{w} \leftarrow \mathbf{w} + \eta((1-\alpha)\varepsilon_{ui}\mathbf{r}_{ui} - \lambda\mathbf{w}) \tag{13}$$

$$\mathbf{w}_u \leftarrow \mathbf{w}_u + \eta((1-\alpha)\varepsilon_{ui}\mathbf{r}_{ui} - \lambda\mathbf{w}_u) \tag{14}$$

$$\mathbf{s}_u \leftarrow \mathbf{s}_u + \eta(\alpha(1-\beta)\varepsilon_{ui}\mathbf{v}_i - \lambda\mathbf{s}_u) \tag{15}$$

$$\mathbf{p}_u \leftarrow \mathbf{p}_u + \eta(\alpha\beta\varepsilon_{ui}\mathbf{q}_i - \lambda\mathbf{p}_u) \tag{16}$$

$$\mathbf{q}_i \leftarrow \mathbf{q}_i + \eta(\alpha\beta\varepsilon_{ui}\mathbf{p}_u - \lambda\mathbf{q}_i) \tag{17}$$

Here, $\varepsilon_{ui} = v_{ui} - (1-\alpha)(\mathbf{w}+\mathbf{w}_u)^T\mathbf{r}_{ui} - \alpha\left[(1-\beta)\sum_{u'\in C} s_{uu'}v_{u'i} + \beta\mathbf{p}_u\mathbf{q}_i^T\right]$, η is learning rate. After getting α, $\mathbf{w}$, $\mathbf{w}_u$, $\mathbf{s}_u$, $\mathbf{p}_u$, and $\mathbf{q}_i$, given $R2 = \{r_{T1+1}, r_{T1+2}, \ldots, r_T\}$, we can use $\hat{v}_{ui} = (1-\alpha)(\mathbf{w}+\mathbf{w}_u)^T\mathbf{r}_{ui} + \alpha\left[(1-\beta)\sum_{u'\in C} s_{uu'}v_{u'i} + \beta\mathbf{p}_u\mathbf{q}_i^T\right]$ to predict the review rating.

4. Experiments and Evaluations

We conducted experiments on four datasets that provide user review and rating information in order to evaluate our proposed model. Section 4.1 described the datasets and the evaluation metric in our experiments. Section 4.2 introduced the experimental settings and research problem. Section 4.3 discussed the performance evaluation. Sections 4.4 and 4.5 described the parameters and the influencing factors of our model, respectively.

4.1. Datasets and the Evaluation Metric

In order to verify the performance of our proposed methods, we performed some experiments on two English datasets and two Chinese datasets. The two public English datasets are from Yelp2013 and Yelp2014, which is a large-scale dataset consisting of restaurant reviews (https://www.yelp.com/

dataset/challenge). At the same time, in order to evaluate the performance of our model in Chinese reviews, we constructed two Douban movie review datasets because there is no suitable public dataset in Chinese.

Douban is a popular Chinese website. Users can post comments on movies, books, and music and at the same time, give a 1–5 star rating. We first download the Douban movie user information through the Application Programming Interface (API) provided by Douban and then sort the Douban movie users according to the number of reviews published. We choose users who have published more than 50 movie reviews as seed users. We obtain the movie reviews published by seed users through the Douban API interface. Based on the captured Douban movie review data, two movie review datasets were constructed. Table 2 shows the statistical information on the four datasets.

Table 2. Statistical information of Yelp2014, Yelp2013, and two Douban movie review datasets.

Datasets	#users	#reviews	#items	#reviews/user	Matrix Density
Douban1	1476	22593	3041	15.31	0.005034
Douban2	1079	13858	2087	12.84	0.006154
Yelp2014	4818	231163	4194	47.97	0.011440
Yelp2013	1631	78966	1633	48.42	0.029648

In Table 2, the user-item rating matrix density is calculated as follows.

$$\text{Density}_{\text{UIRM}} = \frac{\text{Number(reviews)}}{\text{Number(users)} \times \text{Number(items)}} \tag{18}$$

Mean absolute error (MAE) and root mean square error (RMSE) are used as metrics to evaluate the performance of RRP methods. MAE and RMSE are defined as follows:

$$\text{MAE} = \frac{\sum\limits_{testsets} |\hat{v}_{ui} - v_{ui}|}{N_{\text{total}}} \tag{19}$$

$$\text{RMSE} = \sqrt{\frac{\sum\limits_{testsets} (\hat{v}_{ui} - v_{ui})^2}{N_{\text{total}}}} \tag{20}$$

Here, $\hat{v}_{ui}$ is the predicted score by various methods, v_{ui} is the true score of the review in the test set, and N_{total} is the reviews number in the test set.

4.2. Experimental Settings and Research Questions

We randomly split each dataset into two parts; training datasets and test datasets. A total of 80% of each dataset is used as the training dataset and the rest is used as the test dataset. All the hyper-parameters in our methods are selected in the training dataset. We compare our proposed models to several baseline methods.

RRP+LR: RRP method based on review text content by using a linear regression model.

RRP+KNN: RRP method using *k*-nearest neighbor based on user similarity.

RRP+MF: RRP method using matrix factorization.

UPRRP+UPRM: UPRRP method based on the user-item rating matrix by combining *k*-nearest neighbor and matrix factorization.

UPRRP+RTC: UPRRP method based on review text content by modeling the sentiment commonality of different users and sentiment personality of an individual user.

UPRRP+RTC+UPRM: UPRRP method based on review text content information and user-item rating matrix information.

By combining the review text content information and user-item rating matrix information, we propose a UPRRP method based on the review text content and the user-item rating matrix.

To analyze the performance of our method and the factors that affect the performance of our method, we performed three experiments to answer the following three questions in four different datasets.

(1) Whether the performance of our method is better than the benchmark method.

(2) The sensitivity of our method to parameters α and β.

(3) Analysis of factors affecting the performance of our methods.

4.3. Performance Comparison of Different Methods

In this subsection, we compared our method and three benchmark methods on four different datasets. The RRP results of six different methods are shown in Table 3.

Table 3. Mean absolute error (MAE), root mean square error (RMSE) of six different methods in four datasets.

Datasets	Metric	RRP + KNN	RRP + MF	RRP + LR	UPRRP + UIRM	UPRRP + RTC	UPRRP + RTC + UIRM
Douban1	MAE	1.0659	0.8341	0.8477	0.8125	0.8216	0.8011
Douban1	RMSE	1.4547	1.0653	1.1008	1.0442	1.0491	0.9799
Douban2	MAE	1.0626	0.8056	0.8277	0.7870	0.8081	0.7605
Douban2	RMSE	1.4271	1.0387	1.0741	0.9913	1.0282	0.9794
Yelp2014	MAE	0.7112	0.5132	0.5686	0.4852	0.5158	0.4641
Yelp2014	RMSE	0.9993	0.8146	0.8985	0.8123	0.8326	0.7846
Yelp2013	MAE	0.6987	0.4871	0.5623	0.4762	0.4961	0.4472
Yelp2013	RMSE	0.9856	0.8042	0.8931	0.7914	0.8024	0.7641

From Table 3, we can find that our approach reduced the MAE and RMSE of the RRP and achieved better performance than the three benchmark methods in the four different datasets. Compared with RRP+LR, UPRRP+RTC achieved a better performance. This is because only the sentiment commonality information of different users is considered in RRP+LR, whereas the sentiment commonality of different users and sentiment personality of single users are considered in UPRRP+RTC.

In three baseline methods, RRP+MF have the best performance in four different datasets. Compared with RRP+MF, the MAE of UPRRP+UIRM+RTC separately decreased by 5.6% ((0.8056 − 0.7605) / 0.8056) and 8.2% ((0.4871 − 0.4472) / 0.4871) in Douban2 datasets and Yelp2013 datasets. Experimental results in four different datasets proved that UPRRP+UIRM+RTC can improve the performance of the RRP. This is because the user's personalized information is considered in UPRRP+UIRM+RTC by combining the review content and the user-item rating matrix information.

4.4. Parameter Analysis

We study the effects of different parameter settings on the performance of our proposed UPRRP method in this section. In our method, there are two parameters α and β that need to be set. According to the discussion in Section 3, there is a sequence problem in setting the two parameters. The parameter β is determined first, and then the parameter α is determined.

On the training dataset, we use 10-fold cross-validation to obtain the optimal parameters α and β. The MAE of UPRRP+UIRM vary with the parameter β in four different datasets, as shown in Figure 1. After getting the optimal parameter β, we obtained the optimal parameter α by 10-fold cross-validation in train datasets. The MAE of UPRRP+UIRM+RTC vary with the parameter α in four different datasets, as shown in Figure 2.

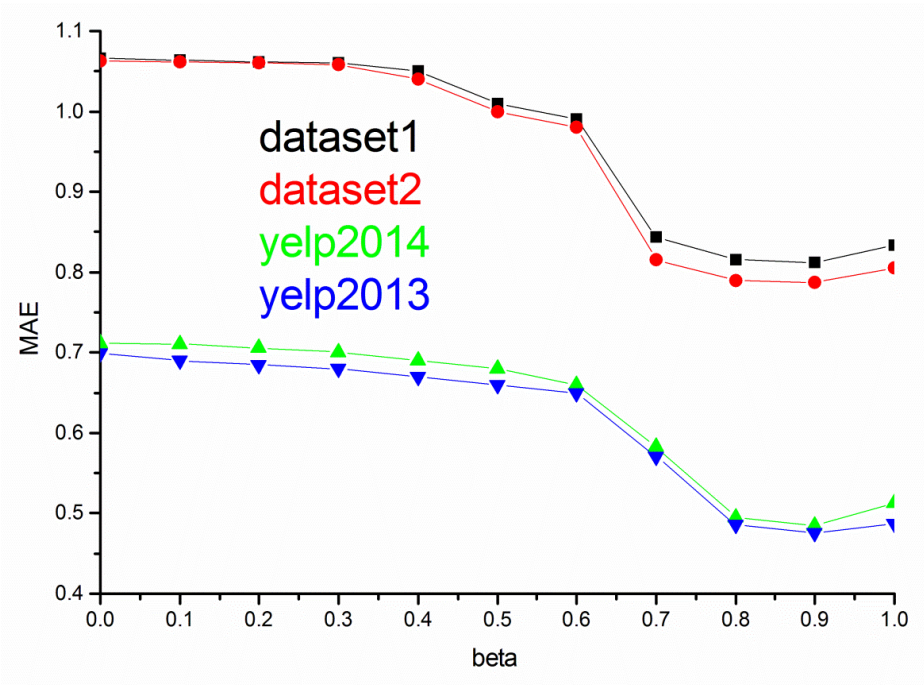

Figure 1. The MAE of UPRRP+UIRM vary with the parameter β in four different datasets. UPPRP = user-personalized review rating prediction; UIRM = user-item rating matrix

In Figure 1, when the parameter β = 0.9, the MAE of the UPRRP+UIRM are the smallest. The reason for this is that the user-item rating matrix for the four datasets is very sparse. Existing research shows that a KNN collaborative filtering algorithm has poor performance compared to a MF collaborative filtering algorithm in the sparse user-item rating matrix [27,28]. The results of the four different datasets also yield the same conclusion: Compared with the KNN collaborative filtering algorithm, the collaborative filtering algorithm based on matrix decomposition has a better performance. The parameter β represents the ratio of the two collaborative filtering algorithms of KNN and MF. Therefore, when the parameter β is selected to be a larger value, the performance of the UPRRP+UIRM method should be better.

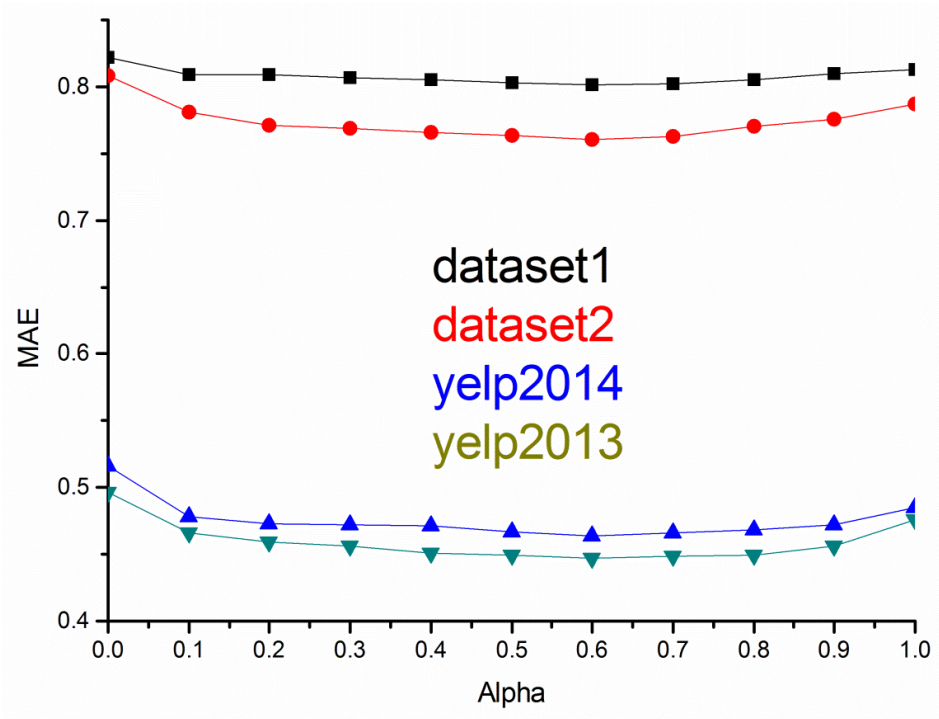

Figure 2. The MAEs of UPRRP+UIRM+RC vary with the parameter alpha in four different datasets.

In Figure 2, when we discuss the parameters of UPRRP+UIRM+RTC, according to the results of Figure 1, we first fix the parameter β to a value of 0.9. When the parameter α is between 0.5 and 0.7, the MAEs of UPRRP+UIRM+RTC method are the smallest. The reason for this is that when the user-item rating matrix is very sparse, the personalized RRP model based on the review text content has a better performance than the personalized RRP model based on the user-item rating matrix. The parameter α represents the ratio of the two personalized methods based on review text content and the user-item rating matrix. Therefore, when we select a smaller value of the parameter α, the performance of the UPRRP+UIRM+RTC method should be improved. Finally, based on the experimental results and analysis, we set $\alpha = 0.6$ and $\beta = 0.9$.

4.5. The Impact of the User-Item Rating Matrix Density on Our Methods

To evaluate the impact of the user-item rating matrix density on UPRRP, we conducted comparative experiments on two Chinese datasets and two English datasets. The experimental results of the Chinese and English datasets are shown in Figures 3 and 4, respectively.

Compared to Douban2, the user-item rating matrix density of Douban1 is sparser. As can be seen from Figure 3, the MAE of the six different methods in Douban1 is higher than in Douban2. From Figure 4, we can find similar results on the Yelp2013 and Yelp2014. The experimental results from the four different datasets show that the sparser the user-item rating matrix density of the review datasets, the higher the MAE of the RRP, and the worse the performance of the corresponding RRP. This is because the sparse user-item rating matrix density contains less history rating information. When we use less user-item rating matrix information to predict the rating, the performance of our methods is worse.

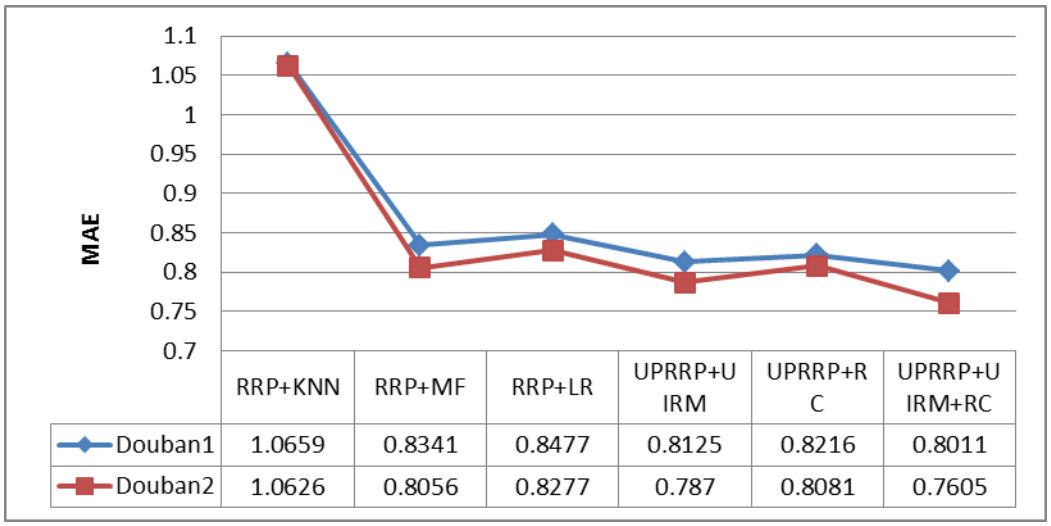

Figure 3. Experimental results in the Chinese Douban 1 and Douban 2.

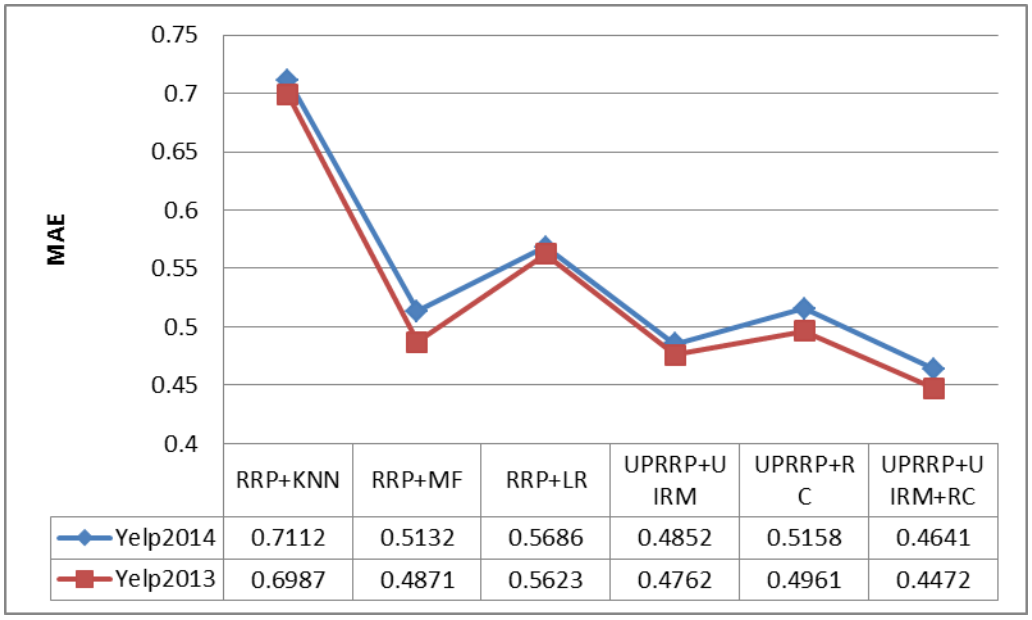

Figure 4. Experimental Results in the English Yelp2014 and Yelp2013.

5. Conclusions

In this paper, we present a novel UPRRP method based on the review text content and user-item rating matrix. To be specific, in order to solve the problem of existing RRP methods based on review text content, we firstly model the commonality and personality of the user's sentiment expression based on the review text content. Secondly, considering that the user-personalized information can be obtained not only from the review text content but also from the user-item rating matrix, we propose a UPRRP method based user-item rating matrix to achieve user-personalized modeling. Finally, we linearly integrate the review text content and the user-item rating matrix information to achieve UPRRP. Experimental results on four datasets show that our proposed methods have better performance than the state-of-the-art baselines in RRP. In the future, we will further model users' personality sentiment expression by deep neural network methods.

Author Contributions: conceptualization, B.W.; methodology, B.W.; software, B.C.; validation, B.C.; formal analysis, B.W.; investigation, B.W.; data curation, G.Z.; writing—original draft preparation, B.W.; writing—review and editing, B.W., Project administration, L.M.

Funding: This work was supported in part by National Natural Science Foundation of China (61472092), Foundation of He'nan Science Technology Committee (172102210428), Foundation of He'nan Educational Committee (19A520032) and Ph.D. Start-up Foundation of Pingdingshan University (PXY-BSQD-2018007).

Acknowledgments: The authors would like to thank all anonymous reviewers and editors for their helpful suggestions for the improvement of this paper.

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

References

1. Piryani, R.; Madhavi, D.; Singh, V.K. Analytical mapping of opinion mining and sentiment analysis research during 2000–2015. *Inf. Process. Manag.* **2016**, *53*, 122–150. [CrossRef]
2. Liu, B. Sentiment analysis and opinion mining. *Synth. Lect. Hum. Lang. Technol.* **2012**, *5*, 1–167. [CrossRef]
3. Khan, F.H.; Qamar, U.; Bashir, S. A semi-supervised approach to sentiment analysis using revised sentiment strength based on Senti Word Net. *Knowl. Inf. Syst.* **2017**, *51*, 851–872. [CrossRef]
4. Khan, F.H.; Qamar, U.; Bashir, S. e SAP: A decision support framework for enhanced sentiment analysis and polarity classification. *Inf. Sci.* **2016**, *367*, 862–873. [CrossRef]
5. Khan, F.H.; Qamar, U.; Bashir, S. Multi-objective model selection (MOMS)-based semi-supervised framework for sentiment analysis. *Cognit. Comput.* **2016**, *8*, 614–628. [CrossRef]
6. Khan, F.H.; Qamar, U.; Bashir, S. SWIMS: Semi-supervised subjective feature weighting and intelligent model selection for sentiment analysis. *Knowl.-Based Syst.* **2016**, *100*, 97–111. [CrossRef]
7. Kiritchenko, S.; Zhu, X.; Mohammad, S.M. Sentiment analysis of short informal texts. *J. Artif. Intell. Res.* **2014**, *50*, 723–762. [CrossRef]
8. Horrigan, J. "Online shopping," Pew Internet and American Life Project Report. *Pew Research Center*. 2008. Available online: http://www.pewinternet.org/2008/02/13/online-shopping/.
9. Wu, Y.; Ester, M. FLAME: A probabilistic model combining aspect based opinion mining and collaborative filtering. In Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, Shanghai, China, 2–6 February 2015; pp. 199–208.
10. Qu, L.; Ifrim, G.; Weikum, G. The bag-of-opinions method for review rating prediction from sparse text patterns. In Proceedings of the 23rd International Conference on Computational Linguistics, Beijing, China, 23–27 August 2010; pp. 913–921.
11. Li, F.; Liu, N.; Jin, H.; Zhao, K.; Yang, Q.; Zhu, X. Incorporating reviewer and item information for review rating prediction. In Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, Barcelona, Spain, 16–22 July 2011; Volume 11, pp. 1820–1825.
12. Ganu, G.; Elhadad, N.; Marian, A. Beyond the Stars: Improving Rating Predictions using Review Text Content. In Proceedings of the Twelfth International Workshop on the Web and Databases, WebDB, Providence, RI, USA, 28 June 2009; Volume 9, pp. 1–6.
13. Zheng, L.; Noroozi, V.; Yu, P.S. Joint deep modeling of users and items using reviews for recommendation. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, Cambridge, UK, 6–10 February 2017; pp. 425–434.
14. 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, Washington, DC, USA, 24–28 July 2010; pp. 783–792.
15. Wu, F.; Huang, Y. Personalized Microblog Sentiment Classification via Multi-Task Learning. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI-16), Phoenix, AZ, USA, 12–17 February 2016; pp. 3059–3065.
16. Shi, Y.; Larson, M.; Hanjalic, A. Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACM Comput. Surv.* **2014**, *47*, 3. [CrossRef]
17. Ma, H. An experimental study on implicit social recommendation. In Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, Dublin, Ireland, 28 July–1 August 2013; pp. 73–82.
18. Koren, Y.; Bell, R.; Volinsky, C. Matrix factorization techniques for recommender systems. *Computer* **2009**, *42*, 30–37. [CrossRef]

19. Koren, Y. Collaborative filtering with temporal dynamics. *Commun. ACM* **2010**, *53*, 89–97. [CrossRef]
20. Colace, F.; De Santo, M.; Greco, L.; Moscato, V.; Picariello, A. A collaborative user-centered framework for recommending items in Online Social Networks. *Comput. Hum. Behav.* **2015**, *51*, 694–704. [CrossRef]
21. Yu, K.; Zhu, S.; Lafferty, J.; Gong, Y. Fast nonparametric matrix factorization for large-scale collaborative filtering. In Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Boston, MA, USA, 19–23 July 2009; pp. 211–218.
22. Li, P.; Wang, Z.; Ren, Z.; Bing, L.; Lam, W. Neural rating regression with abstractive tips generation for recommendation. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Tokyo, Japan, 07–11 August 2017; pp. 345–354.
23. Pang, B.; Lee, L. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, Ann Arbor, MI, USA, 25–30 June 2005; pp. 115–124.
24. Liu, J.; Seneff, S. Review sentiment scoring via a parse-and-paraphrase paradigm. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, Singapore, 6–7 August 2009; Volume 1, pp. 161–169.
25. Lee, H.C.; Lee, S.J.; Chung, Y.J. A study on the improved collaborative filtering algorithm for recommender system. In Proceedings of the 5th ACIS International Conference on Software Engineering Research, Management & Applications (SERA 2007), Busan, Korea, 20–22 August 2007; pp. 297–304.
26. Jeong, B.; Lee, J.; Cho, H. Improving memory-based collaborative filtering via similarity updating and prediction modulation. *Inf. Sci.* **2010**, *180*, 602–612. [CrossRef]
27. Covington, P.; Adams, J.; Sargin, E. Deep neural networks for youtube recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, 15–19 September 2016; pp. 191–198.
28. He, X.; Chua, T.S. Neural factorization machines for sparse predictive analytics. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Tokyo, Japan, 7–11 August 2017; pp. 355–364.
29. Catherine, R.; Cohen, W. TransNets: Learning to Transform for Recommendation. *arXiv*, 2017; arXiv:1704.02298.
30. Kim, D.; Park, C.; Oh, J.; Lee, S.; Yu, H. Convolutional Matrix Factorization for Document Context-Aware Recommendation. In Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, 15–19 September 2016; ACM: New York, NY, USA, 2016.
31. Seo, S.; Huang, J.; Yang, H.; Liu, Y. Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In Proceedings of the Eleventh ACM Conference on Recommender Systems, Como, Italy, 27–31 August 2017; ACM: New York, NY, USA, 2017; pp. 297–305.
32. He, X.; Chen, T.; Kan, M.Y.; Chen, X. Trirank: Review-aware explainable recommendation by modeling aspects. In Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, Melbourne, Australia, 18–23 October 2015; ACM: New York, NY, USA, 2015; pp. 1661–1670.
33. Ling, G.; Lyu, M.R.; King, I. Ratings meet reviews, a combined approach to recommend. In Proceedings of the 8th ACM Conference on Recommender systems, Foster City, CA, USA, 6–10 October 2014; pp. 105–112.
34. 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, Hong Kong, China, 12–16 October 2013; pp. 165–172.
35. Ren, Z.; Liang, S.; Li, P.; Wang, S.; de Rijke, M. Social collaborative viewpoint regression with explainable recommendations. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, Cambridge, UK, 6–10 February 2017.
36. Bao, Y.; Fang, H.; Zhang, J. Topicmf: Simultaneously exploiting ratings and reviews for recommendation. In Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, Québec City, QC, Canada, 27–31 July 2014; pp. 2–8.
37. Diao, Q.; Qiu, M.; Wu, C.; Smola, A.J.; Jiang, J.; Wang, C. Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS). In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 24–27 August 2014; pp. 193–202.
38. Jakob, N.; Weber, S.H.; Müller, M.C.; Gurevych, I. Beyond the stars: Exploiting free-text user reviews to improve the accuracy of movie recommendations. In Proceedings of the 1st International CIKM Workshop on Topic-Sentiment Analysis for Mass Opinion, Hong Kong, China, 6 November 2009; pp. 57–64.

39. Zhang, W.; Yuan, Q.; Han, J.; Wang, J. Collaborative Multi-Level Embedding Learning from Reviews for Rating Prediction. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, New York, NY, USA, 9–15 July 2016; pp. 2986–2992.
40. Zhang, Y.; Ai, Q.; Chen, X.; Croft, W.B. Joint representation learning for top-n recommendation with heterogeneous information sources. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, Singapore, 6–10 November 2017; pp. 1449–1458.
41. Gong, L.; Al Boni, M.; Wang, H. Modeling social norms evolution for personalized sentiment classification. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; Volume 1, pp. 855–865.