Rating Prediction via Exploring Service Reputation

Xiaojiang Lei, Xueming Qian

SMILES LAB of Xi'an Jiaotong University, Xi'an CN710049, China

xiaolei3439@stu.xjtu.edu.cn; qianxm@mail.xjtu.edu.cn

***Abstract*—With the explosion of e-commerce, it presents a great opportunity for people to share their consumption experience in review websites. However, at the same time we face the information overloading problem. How to mine valuable information from these reviews and make an accurate recommendation is crucial for us. Traditional recommender systems (RS) consider many factors, such as product category, geographic location, user's purchase records, and the other social network factors. In this paper, we firstly propose a social user's reviews sentiment measurement approach and calculate each user's sentiment score on items/services. Secondly, we consider service reputation, which reflects the customers' comprehensive evaluation. At last, we fuse service reputation factor into our recommender system to make an accurate rating prediction, which is based on probabilistic matrix factorization. We conduct a series of experiments on Yelp dataset, and experimental results show the proposed approach outperforms the existing RS approaches.**

I. INTRODUCTION

As we all know, service reputation is important for customer to make decisions, which reflects consumers' comprehensive evaluation based on the intrinsic value of a specific product. If we want to know service reputation, users' textual reviews are necessary. In our daily life, users are most likely to buy those items that are posted with highly praise reviews. Hence, how to mine reviews information to recommend user favourite and satisfying items has become an important issue in web mining, machine learning and natural language processing.

Extracting users' interests with the content of reviews has received considerable attention in recent years. The rise like Douban[1], Yelp[2] and other review websites has provided a broad thought in mining users' preferences and predicting users' ratings. Jiang et al. [19] propose an author topic model-based collaborative filtering (ATCF) method, which facilitates comprehensive points of interest (POIs) recommendations for social users. We observe that in many practical cases, it is more important to provide numerical ratings rather than binary decisions. Especially when a customer compares several candidate products, all of them reflect positive sentiment in a binary classification. To make a purchase decision, customers not only need to know whether the product is good or not, but also how good the product is [7]. In our daily life, when we search the net for purchasing, both positive reviews and negative reviews are valuable to be as reference. For positive

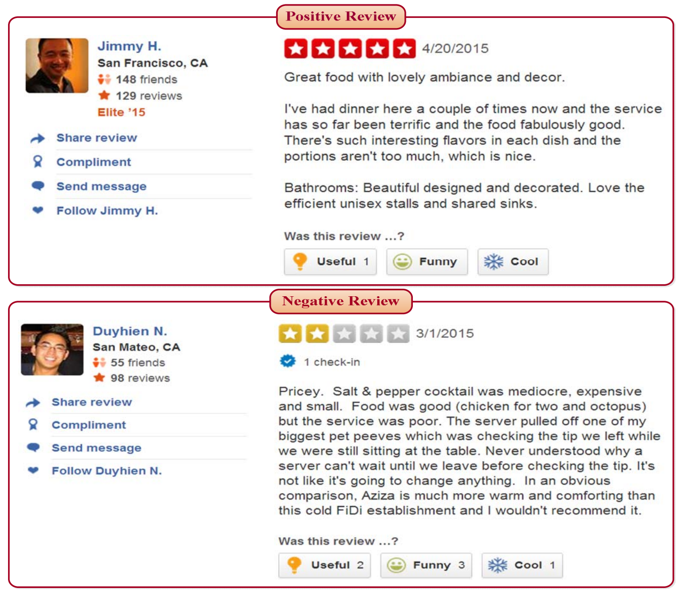

Fig. 1. An example of positive review and negative review on Yelp website.

reviews, we can know the advantages of the product. For negative reviews, we can obtain the shortcomings in case of being cheated. In Fig.1, we intuitively show an example of positive reviews and negative reviews on Yelp website. From Fig.1, there are many positive words in a 5-star review, such as "*great*", and "*lovely*". But in a 2-star review we find negative words, such as "*expensive*", and "*poor*". That means a good review reflects a high star-level and a bad review reflects a low-level. When we know the advantages and disadvantages from the two kinds of reviews, we can easily make a decision. Normally, if item's reviews reflect positive sentiment, then the item may be with good reputation. Oppositely, if item's reviews are full of negative sentiment, then the item is most likely with bad reputation. So based on users' reviews sentiment, we can infer users' comprehensive ratings on items. However, users' sentiment is hard to predict and the unpredictability of service reputation makes a great difficulty in exploring social users.

To address these problems, we propose a rating prediction model, which makes the best use of users' sentiment extracted from the textual descriptions of users' reviews/comments. When compared with previous works [1-3], [6], [7], [17]-[22] the main difference is that: previous works mainly focus on exploring social user's salient feature and discussing how to classify sentiment, and do not go deeper in mining users' sentiment for potential applications. But in our paper, we adequately mine user's sentiment, and leverage the sentiment to infer service reputation which proved to make great contributions to the prediction accuracy of RS.

2015 IEEE 17th International Workshop on Multimedia Signal Processing (MMSP), Oct 19-21, 2015, Xiamen, China.
978-1-4673-7478-1/15/$31.00 ©2015 IEEE

[1] http://www.douban.com
[2] http://www.yelp.com

The main contributions of our approach are summarized as follows: 1) We propose a social user's sentiment measurement approach based on the mined sentiment words and sentiment degree words from users' reviews. 2) We build item's "Virtual Friends" to solve the cold start problem. Then we leverage "Virtual Friends" to calculate service reputation similarity to help predict users' ratings. 3) We take service reputation factor into a probabilistic matrix factorization model to carry out an accurate recommendation. The experimental results and discussions show that the service reputation factor is effective to improve rating prediction performances.

The rest of this paper is organized as follows. We firstly give an overview of related work in Section II. We then describe our detailed approach in Section III. Finally, we report the experimental results and analysis in Section IV and provide a conclusion in Section V.

II. Related Work

In this section, we survey recent works related to our approach. We firstly review some classical approaches based on matrix factorization, which have been used for recommendation. Then the reviews based approaches in social networks are briefly reviewed. Finally, sentimental mining and applications are provided.

A. Matrix Factorization Based Approaches

1) Basic Probabilistic Matrix Factorization

Matrix factorization is one of the most popular approaches for low-dimensional matrix decomposition. Here, we review the Base MF [1]. The rating matrix $R \in \boldsymbol{R}^{m\times n}$ (m is the number of users and n is the number of items) can be predicted according to Eq. (1), where $U_u \in \boldsymbol{U}^{\mathrm{m\times k}}$ denotes the user Potential Eigen vectors matrix and $P_i \in \boldsymbol{P}^{n\times k}$ denotes item Potential Eigen vectors matrix, and k is the dimension of the vectors. $\bar{R}$ denotes the average score of all ratings.

$$\hat{R}_{u,i} = \bar{R} + U_u P_i^T \tag{1}$$

The task of this model is to reduce the error of predicted rating values to real rating values. Normally, items have five rating star levels (1-5 stars). And the sum-of-squared-error objective function Ψ is defined (with Frobenius regularization $\|.\|_F$).

$$\Psi(\boldsymbol{R},\boldsymbol{U},\boldsymbol{P}) = \frac{1}{2}\sum_{u,i}(R_{u,i} - \hat{R}_{u,i})^2 + \frac{\lambda}{2}(\|U\|_F^2 + \|P\|_F^2) \tag{2}$$

where $\hat{R}_{u,i}$ denotes predicted objective star level evaluation of item i. The optimization problem of the objective function Ψ can be solved using gradient descent method.

2) Social Recommendation

In real life, people's decision is often affected by friends' action or recommendation. How to utilize social information has been extensively studied. Yang et al. [2] propose the concept of "Trust Circles" in social network based on probabilistic matrix factorization. And CircleCon model [2] has been found to outperform Base MF[1] and SoRec [15] with respect to accuracy of the RS. The trust value between users is represented by the matrix $\boldsymbol{S}$, and directed and weighted social relationship of user u with user v is represented by a positive value $S_{u,v}^{c*} \in [0,1]$. The basic idea is that user latent feature U_u should be similar to the average of his/her friends' latent features with weight of $S_{u,v}^{c*}$ in category c. Once the model is trained in c, the rating value in category c can be predicted according to Eq. (1). Qian et al. [3] and Feng [22] propose a personal recommender model (**PRM**) combing with several social factors. They make full use of categories of goods, and user personal interest is the main contributions of the approach. Zhao et al.[17] propose a unified model to evaluate services by exploring user ratings confidence. They utilize entropy to evaluate user ratings confidence and take spatial and temporal features into consideration. They improve the accuracy of predicting, but all of these methods don't take advantages of users' reviews on the website. At the same time, there also remain a few questions: some users may have no social relation with each other or even worse, explicit social networks information is not always available and it is difficult to provide good prediction for every users or items. In this paper, we elaborate to use service reputation and "virtual friends" to improve social recommendation.

B. Reviews Based Applications

There are also some textual reviews based works for the task of recommendation. Feng et al. [6] use social images and tagged text information and they propose a novel hierarchical user interest mining approach for personalized products recommendation. Ling et al. [9] propose a unified model that combines content-based filtering with collaborative filtering, and harnessing the information of both ratings and reviews. Moreover, they apply topic modeling techniques to improve prediction accuracy. Aiming at accurate user classification for tag application systems, Zhao et al. [10] mine users' intention in reviews and extend the tag semantics by open knowledge platform. Their research can not only be used as the basis of the users' interests and preferences research, but also can be employed in non-Tag application. Xu et al. [11] propose a new personalized recommendation model, i.e. topic model based collaborative filtering (TMCF) utilizing users' reviews and ratings. They exploit extended Latent Dirichlet Allocation (LDA) model to generate topic allocations for each review and then obtain each user's preferences.

C. Sentiment Based Applications

There are many approaches around sentiment analysis and rating prediction. Zhang et al. [4] fuse the self-supervised emotion integrated sentiment classification results into collaborative filtering (CF) recommenders. This model greatly solved the rating prediction problems for those users who have no ratings but a list of reviews. Based on reviews factor, Phan et al. [5] propose a content-based ranking method in which the user engagement and the comment polarity are all considered. In their paper, they analyse users' comment by using a lexicon based approach. Zhang et al. [14] propose the Explicit Factor Model (EFM). They extract explicit product features and user opinions by phrase-level sentiment analysis on user reviews, then generate both recommendations and dis-recommendations according to the specific product features to the user's interests and the hidden features are also learned.

III. The Approach

We propose a recommender model, which considers service reputation factor. Our idea is to make full use of users' subjective sentiment of the items, which can be explored from users' textual reviews. In order to better understand the approach, we firstly give the calculation method of sentiment, and then we describe how to use sentiment information to infer service reputation, at last we fuse service reputation factor into rating prediction model, which is based on matrix factorization. Hereinafter, we turn to the detail of our approach.

A. Identifying User's Sentiment

In this section, we build three sentiment dictionaries (**SD**, **SDD**, and **ND**) based on HowNet Sentiment Dictionary[3] to calculate users' sentiment on items, which is well known in the area of Chinese and English sentiment classification.

The sentiment dictionary (**SD**) has 8938 word items in total, which contains 4363 positive sentiment words and 4575 negative sentiment words. For example, the words "*attractive*", "*brilliant*", and "*convenient*" belong to positive sentiment words, and they represent positive sentiment. Meanwhile, the words "*annoyed*", "*boring*", and "*unfortunate*" belong to negative sentiment words, and they represent negative sentiment.

The sentiment degree dictionary (**SDD**) has 128 words in total. We classify the **SDD** into five different levels of degree, and some example words are shown in Table 1. There are 52 words in the **Level-1,** which mean the highest degree of emotion, such as the words "*most*", "*best*", and "*greatest*". And 48 words in the **Level-2**, which mean higher degree of emotion, such as the words "*better*", "*over*", and "*very*". There are 12 words in the **Level-3**, such as the words "*more*", "*even*", and "*such*". There are 9 words in the **Level-4,** such as the words "*a little*", "*a bit*", and "*more or less*". And there are 7 words in the **Level-5**, which mean lowest degree of emotion, such as the words "*less*", "*bit*", and "*not very*".

The negation dictionary (**ND**) contains 56 negative prefix words in total, such as "*no*", "*hardly*", and "*can not*". These words are used to judge whether there has need to reverse the emotion polarity of sentiment words.

TABLE 1. RULE BASED DEGREE DETERMINATION FOR SENTIMENTAL DEGREE WORDS

Level	D_w	**Sentiment Degree Words**
1	5	Most, best, greatest, absolutely, highly…
2	4	Over, very, greatly, much, *really*, super…
3	2	Even, more, far, so, *such*, intensely…
4	0.5	A little, a bit, rather, more or less…
5	0.25	Less, not very, bit, little…

For each user's original review on item, we firstly remove some unnecessary words using stop words list[4], including prepositions, articles, conjunctions and pronouns. And then we filter out the noise data such as advertisement and digital gibberish. After filtering out all noise words, each textual review input will be divided into several clauses by the punctuation mark. Then for each clause, firstly, we find out the sentiment words based on the dictionary **SD**. A positive word is initially assigned with score +1.0, while a negative word is assigned with score −1.0. Secondly, we find out the sentiment degree words based on the dictionary **SDD** and take the sentiment degree words into consideration as a weight of the found sentiment words. Finally, we find out the negative prefix words based on the dictionary **ND** and add a negation check coefficient that has a default value of +1.0. If the sentiment word is preceded by a negation within the specified zone, this coefficient is set to −1.0. Then for a review *r* that user *u* posts for the item *i*, we get the sentiment score as follows:

$$S(r) = \frac{1}{N_c}\sum_{c\in r}\sum_{w\in c} Q \cdot D_w \cdot R_w \tag{3}$$

where *c* denotes the clause. N_c denotes the number of clauses. Q denotes the negation check coefficient. D_w is determined by a simple rule based approach in Table 1, and there is a one-to-one correlation between D_w and five sentimental degree levels. R_w denotes the initial score of the sentiment word *w*.

After we obtained the review *r*'s basic sentiment score, we normalize the score as follows:

$$E_{u,i} = \frac{10}{1+e^{-S(r)}} - 5 \tag{4}$$

where $E_{u,i}$ denotes user *u*'s sentiment score on item *i*, *S*(*r*) is the basic sentiment score calculated in Eq.(3).

Fig. 2. An example of review analysis for identifying user's sentiment on Yelp website.

Intuitively, we analyse a real user's review on Yelp website and give the processing result as shown in Fig. 2. In Fig. 2, we can see that the user's original review is divided into three clauses (N_c=3). And each clause only retains the more important words. In clause1, "*great*" is a positive sentiment word ($R_w = 1$), "***such***" is a Level-3 sentiment degree word (D_w=2). In clause2, both of the words "*friendly*" and "*reasonable*" are positive sentiment words ($R_w = 1$), and "***really***" is a Level-2 sentiment degree word (D_w=4). In clause3, "*small*" is a negative word ($R_w = -1$), "*tidy*", "*delicate*", and "*tasty*" are all positive words ($R_w = 1$). At the

---

[3] http://www.keenage.com/download/sentiment.rar
[4] http://www.ai.mit.edu/projects/jmlr/papers/volume5/lewis04a/

same time, the word "***really***" is a Level-2 sentiment degree word (D_w=4). There are no negative words in three clauses. To sum up all of the three clauses, we get user u's sentiment $S(r) = \frac{1}{3}\{(1\times2)+(1\times4+1)+[(-1)+1+1+1\times4]\} = 4$. After normalizing the basic score, we have $E_{u,i} = \frac{10}{1+e^{-S(r)}} - 5 \approx 4.82$. Based on this method, we can get all users' sentiment score.

B. Service Reputation Factor

From typical item-based collaborative filtering algorithms [13], we can know that similar items can help predict item's ratings. So it's important for us to find those items that have similar features. In our paper, we think service reputation can indirectly reflect its real rating score, and we leverage users' sentimental distribution to infer service reputation (hereinafter, item represent service and they have same meanings). Based on users' sentiment, we also think that if two items have similar sentimental distribution, then they may have similar reputation, and mostly they will be posted with similar ratings.

Based on the idea, we assume that the user set is $U=\{u_1, u_{2,\cdots,}u_M\}$, where M is the number of users. After obtaining each item's normalized sentiment score $E_{u,i}$ in Eq.(4), we use items' sentimental distribution among all users to denote item i's reputation $W_i = \{E_{u_1,i}, E_{u_{2,},i}, \cdots, E_{u_M,i}\}$, where $E_{u,i}$ denotes user u's sentiment score on item i. Then we choose some items as "Virtual Friends" of the item, which have been rated by the same users. After that, we calculate the reputation similarity between the item i and its' virtual friend j. Here we hold that item's latent feature P_i should be similar to its friends' latent feature with the weight of $I_{i,j}$. Then we use cosine similarity to measure the reputation similarity of item i and item j as follows:

$$I_{i,j} = cosine(\, W_i, W_j \,) \tag{5}$$

In order to fuse items' reputation similarity factor into matrix factorization model, we normalize $I_{i,j}$ as follows:

$$I^*_{i,j} = \frac{I_{i,j}}{\sum_{j \epsilon F_i} I_{i,j}} \tag{6}$$

where F_i denotes the set of item i's virtual friends, and "*" is a normalized symbol, and each of the rows is normalized to unity $\sum_{j\epsilon F_i} I^*_{i,j} = 1$.

C. Our Recommender Model

After taking the items' reputation factor above into consideration, we get an important constrain term in our rating prediction model, normalized service reputation similarity $I^*_{i,j}$, which is calculated by Eq.(6). The sum-of-squared-error objective function Ψ is defined as

$$\begin{aligned}&\Psi(\boldsymbol{R}, \boldsymbol{U}, \boldsymbol{P})\\ &= \frac{1}{2}\sum_{u,i}\left(\hat{R}_{u,i} - R_{u,i}\right)^2 + \frac{\lambda}{2}(\|\boldsymbol{U}\|_F^2 + \|\boldsymbol{P}\|_F^2)\\ &+ \frac{\alpha}{2}\sum_i\left(\left(P_i - \sum_j I^*_{i,j} P_j\right)\left(P_i - \sum_j I^*_{i,j} P_j\right)^T\right)\end{aligned} \tag{7}$$

where $\hat{R}_{u,i}$ is the predicted rating value according to Eq.(8). $R_{u,i}$ is user u's real rating on item i, $\boldsymbol{R}$ is the set of users' ratings on items, and $R_{u,i} \in \mathbf{R}^{M\times N}$, and M is the number of users, N is the number of items. $\boldsymbol{U}^{M\times K}$ and $\boldsymbol{P}^{N\times K}$ denote user Potential Eigen vectors and item Potential Eigen vectors respectively, $\|\mathrm{X}\|_\mathrm{F}$ is the Frobenius norm of matrix **X.** The first term of Eq. (7) denotes deviation between the actual rate and prediction score, the second term of Eq.(7) is a regularization term, which plays a role in case of over-fitting. The idea of the service reputation similarity is enforced by the last term, which says that if two items have the similar sentiment distribution, they may have the similar latent feature P_i.

$$\hat{R}_{u,i} = \bar{R} + U_u P_i^T \tag{8}$$

where $\bar{R}$ is average score of all ratings, U_u and P_i are K-dimensional user-specific and item-specific latent feature vectors of user u and item i, and it is the rank of the latent matrices $\boldsymbol{U}^{M\times K}$ and $\boldsymbol{P}^{N\times K}$, which are obtained by the gradient descent method [3].

D. Model Training

We get the corresponding matrix factorization model as Eq.(7), from which we can obtain user latent profile $\boldsymbol{U}_u$ and item latent profile $\boldsymbol{P}_i$ by using gradient descent method[3]. More formally, the gradients of the objective function with respect to the variables U_u and P_i are shown as (9) and (10) respectively:

$$\frac{\partial\Psi}{\partial U_u} = \sum_i\left(\hat{R}_{u,i} - R_{u,i}\right)P_i + \lambda U_u \tag{9}$$

$$\begin{aligned}\frac{\partial\Psi}{\partial P_i} &= \sum_u\left(\hat{R}_{u,i} - R_{u,i}\right)U_u + \lambda P_i\\ &+\alpha\left(P_i - \sum_{j\epsilon F_i} I^*_{i,j}P_j\right) - \alpha\sum_{i\epsilon F_j} I^*_{j,i}\left(P_j - \sum_{k\epsilon F_j} I^*_{j,k} P_k\right)\end{aligned} \tag{10}$$

where F_i denotes item i's virtual friends while extracting from all items. The initial values of U_u and P_i are sampled from the normal distribution with zero mean. The user and item latent feature vectors U_u and P_i are updated based on the previous values to insure the fastest decreases of the objective function at each iteration. We set the step size ℓ=0.0002 and the iteration number τ =500 to insure the decreases of the objective function while training.

IV. Experiment

In this section, we conduct a series of experiments to evaluate the performance of proposed rating prediction model. We choose Yelp as the main test dataset [3], which contains eight categories: #[1] **Active Life**, #[2] **Beauty&Spa**, #[3] **Home Service**, #[4] **Hotel&Travel**, #[5] **NightLife**, #[6] **Restaurants**, #[7] **Shopping**, and #[8] **pets**. In total, there are 28,629 users, 96,974 items, and 300,847 ratings in our dataset. The compared approaches include Base MF [1], CircleCon [2], and PRM [3].

A. Performance Measures

In each dataset of Yelp, we use 80% of data as the training set and the remaining 20% as the test set. The evaluation metrics we use in our experiments are Root Mean Square Error (RMSE) and Mean Absolute Error (MAE), as they are the most popular accuracy measures in recommendation systems [1], [2], [3]. RMSE and MAE are defined as

$$RMSE = \sqrt{\sum_{(u,i)\in R_{test}}\left(R_{u,i} - \hat{R}_{u,i}\right)^2 / |R_{test}|} \tag{11}$$

where R_{test} is the set of all user-item pairs (u, i) in the test set. MAE is defined as

$$MAE = \sum_{(u,i)\in R_{test}} \left|R_{u,i} - \hat{R}_{u,i}\right| / \left|R_{test}\right| \qquad (12)$$

B. *Evaluation*

1) *Comparative Algorithms*

In this section, we conducted a series of experiments to compare our rating prediction model with the following existing models.

Base MF: This method is the baseline matrix factorization approach proposed in [1], which does not consider any social factors.

CircleCon: This approach proposed in [2], which focuses on the factor of interpersonal trust in the social network and infers the trust circle based on matrix factorization.

PRM: This approach proposed in [3] considers three social factors, including interpersonal influence, interpersonal interest similarity and personal interest based on matrix factorization to predict users' ratings.

OURS: It's our recommender model. We make full use of social users' reviews sentiment to infer service reputation, and then we fuse normalized service reputation similarity into our recommender system. By leveraging service reputation similarity, we can predict users' ratings on items.

2) *Performance Comparison*

In all compared models, K is the dimension of user-specific and item-specific latent feature vectors, λ is a trade-off parameter in case of over-fitting. Here we set parameter K=10, and set λ = 1, which are discussed in detail in [16]. In our objective function, we set $\alpha = 15$, which is a trade-off parameter to adjust the impact of service reputation factor. Note that whatever these parameters are, it is fair for all compared algorithms, because we train all models in the same framework of matrix factorization and set the same parameters in other compared models (i.e. CircleCon, PRM). In Table 2, we show the performance based on the eight Yelp datasets. The percentage numbers in each cell are the relative improvements of our method (sentiment based recommender model) over the various baseline models. It is clearly shown that our recommender model considering service reputation factor outperforms the Base MF, CircleCon and PRM models on each dataset of Yelp. For the baseline approaches, Base MF, CircleCon and PRM model, we decrease the average RMSE by 23.23%, 16.75%, and 4.5%, and we decrease the average MAE by 21.4%, 17.8% and 7.4%. Totally, in a 16GB RAM and i5-3470 CPU computer, it takes 5.24 hours in average of eight Yelp datasets in training our model, which performances efficiently than PRM (6.18 hours) and other models. Because our model uses less factors (i.e. service reputation) and we take advantage of the "Virtual Friends" so that we have no need to calculate all items' reputation similarities. The results in Table 2 demonstrate the importance of social users' sentiment in expressing users' preferences and the factor of service reputation similarity can really improve the accuracy of recommender systems.

C. *Discussion*

Besides the performance comparison of the proposed model with the existing models in Table 2, here we discuss the impact of service reputation similarity by adjusting trade-off parameter α on eight Yelp datasets. To discuss the impact of service reputation similarity, we let α range from 0 to 2000. The experimental results are shown in Fig.3 and Fig.4. From Fig.3 and Fig.4, we can see that the RMSE and MAE drop in all testing categories when α range from 0 to 1000. Besides, when α range from 1000 to 2000, we can see that both of RMSE and MAE increase in different degrees (on **active Life, Home Service, Hotel & travel, and Pets datasets**) because of over-fitting. The average RMSE and MAE values of our model under α=1000 are 1.156 and 0.916 respectively.

TABLE 2. COMPARISONS FOR EIGHT CATEGORIES ON YELP (DIMENSIONALITY K = 10). THE PERCENTAGE NUMBERS IN EACH CELL ARE THE RELATIVE IMPROVEMENTS OF OUR METHOD OVER THE VARIOUS BASELINE MODELS.

DATASETS	Base MF		CircleCon		PRM		OURS	
	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
Active Life	1.633 *27.31%*	1.238 *25.28%*	1.477 *19.63%*	1.126 *17.85%*	1.265 *6.17%*	0.984 *6.0%*	**1.187**	**0.925**
Beauty&Spa	1.813 *25.54%*	1.390 *23.6%*	1.656 *18.48%*	1.272 *16.51%*	1.431 *5.66%*	1.128 *5.85%*	**1.350**	**1.062**
Home Service	1.981 *23.07%*	1.558 *21.25%*	1.844 *17.35%*	1.454 *15.61%*	1.611 *5.40%*	1.284 *4.44%*	**1.524**	**1.227**
Hotel&Travel	1.683 *24.59%*	1.318 *23.44%*	1.539 *17.54%*	1.201 *15.99%*	1.321 *3.94%*	1.042 *3.17%*	**1.269**	**1.009**
Night Life	1.408 *20.95%*	1.099 *19.2%*	1.311 *15.10%*	1.026 *13.45%*	1.150 *3.22%*	0.913 *2.74%*	**1.113**	**0.888**
Pets	1.873 *24.39%*	1.440 *23.13%*	1.715 *17.43%*	1.329 *16.7%*	1.481 *4.39%*	1.181 *6.27%*	**1.416**	**1.107**
Restaurants	1.261 *13.48%*	0.983 *11.29%*	1.202 *9.23%*	0.944 *7.63%*	1.094 *0.27%*	0.873 *0.11%*	**1.091**	**0.872**
Shopping	1.600 *23.19%*	1.228 *21.6%*	1.479 *16.9%*	1.138 *15.38%*	1.302 *5.61%*	1.016 *5.22%*	**1.229**	**0.963**
Average	1.657 *23.23%*	1.282 *21.45%*	1.528 *16.75%*	1.186 *15.09%*	1.332 *4.5%*	1.053 *4.37%*	**1.272**	**1.007**

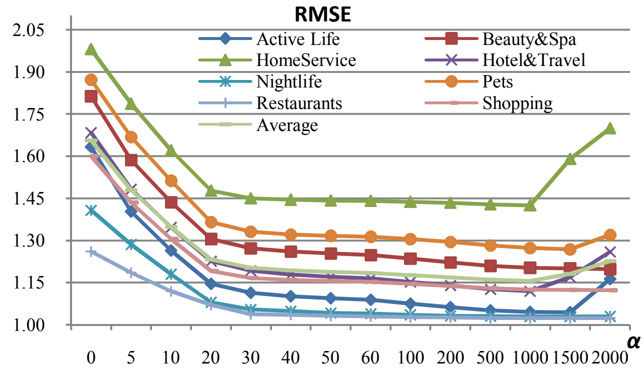

Fig. 3 RMSE line chart of impact of service reputation similarity factor on eight categories of Yelp.

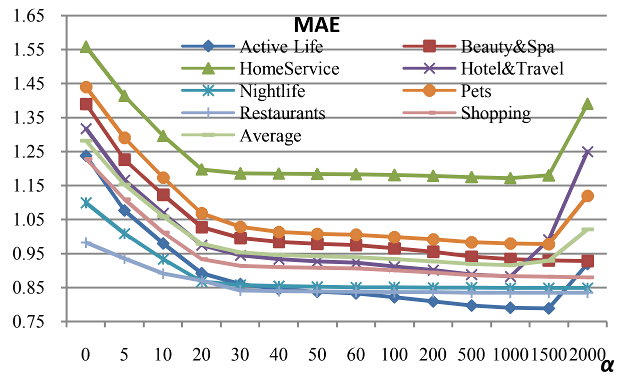

Fig. 4 MAE line chart of impact of service reputation similarity factor on eight categories of Yelp.

Compared with the Base MF [1], we can see that the average error in Yelp datasets decrease about 30.2% and 31.08% respectively. The result of this experiment suggests that the service reputation similarity factor makes great contributions to the accuracy of rating prediction.

V. Conclusion

In this paper, a recommendation model is proposed by mining sentiment information from social users' reviews. We propose social user's sentiment measurement approaches based on the mined sentiment words and sentiment degree words from users' reviews. Besides, we build items' "virtual friends" to solve the cold start problems, and then we fuse service reputation similarity into a unified matrix factorization framework to predict users' ratings. In particular, as long as we extract user's reviews from his/her rating histories, we can quantitively measure user's sentiment. And also we demonstrate users' sentiment can really denote what users' interest preferences. The extensive experiments on real world datasets show significant improvements in rating accuracy over existing approaches. In our future work, we will explore social users' sentiment deeply.

Acknowledgment

This work is supported in part by the Program 973 No.2012CB316400, by NSFC No.60903121, 61173109, 61332018, and Microsoft Research Asia.

References

[1] R. Salakhutdinov, and A. Mnihdepartment, "Probabilistic matrix factorization," in *NIPS,* 2008.

[2] X. Yang, H. Steck, and Y. Liu, "Circle-based recommendation in online social networks," in *proceedings of ACM SIGKDD international conference on Knowledge discovery and data mining*, 2012, pp.1267-1275.

[3] X. Qian, H. Feng, G. Zhao, and T. Mei, "Personalized recommendation combining user interest and social circle," *IEEE Trans. Knowledge and Data Engineering*, vol.26, no.7, 2014, pp.1487-1502.

[4] W. Zhang, G. Ding, L. Chen, C. Li , and C. Zhang, "Generating virtual ratings from Chinese reviews to augment online recommendations," *ACM Transactions on intelligent systems and technology (TIST)*, vol.4, no.1, 2013, pp.1-17.

[5] T. N. Phan, Y. Myungsik, "The lexicon-based sentiment analysis for fan page ranking in Facebook," *IEEE International Conference on Information Networking*, 2014, pp.444-448.

[6] H. Feng, and X. Qian, "Mining user-contributed photos for personalized product recommendation," *Neurocomputing, vol.*129, 2014, pp.409-420.

[7] F. Li, N. Liu, H. Jin, K. Zhao, Q. Yang, X. Zhu, "Incorporating reviewer and product information for review rating prediction," in *proceedings of the Twenty-Second international joint conference on Artificial Intelligence*, 2011, pp.1820-1825.

[8] B. Sun, and V. Ng, "Analyzing sentimental influence of posts on social networks," *IEEE 18th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, 2014, pp.546-551.

[9] G. Ling, Michael R. Lyu, and Irwin King, "Ratings meet reviews, a combined approach to recommend," in *RecSys '14. ACM*, New York, NY, USA, 2014, pp.105-112.

[10] F. Zhao, Q. Qiu, and W. Zhou, "A user classification solution based on users' reviews," *IEEE 13th International Symposium on Distributed Computing and Applications to Business, Engineering and Science*, 2014, pp.243-247.

[11] J. Xu, X. Zheng, W. Ding, "Personal recommendation based on reviews and ratings alleviating the sparsity problem of collaborative filtering," *IEEE International Conference on e-business Engineering*, 2012, pp.9-16.

[12] M. Jiang, P. Cui, F. Wang, W. Zhu, S. Yang, "Scalable recommendation with social contextual information," *IEEE Transactions on Knowledge and Data Engineering*, 2014, pp.2789-2802.

[13] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. WWW*, Hong Kong, 2001, pp.285-295.

[14] Y. Zhang, G. Lai, M. Zhang, "Explicit factor models for explainable recommendation based on phrase-level sentiment analysis," in *Proc. SIGIR'14, ACM*.

[15] H. Ma, H. Yang, M. R. Lyu, and I. King, "SoRec: Social recommendation using probabilistic matrix factorization," in *Proc.17th ACM CIKM,* Napa Vally, CA, USA, 2008, pp.931-940.

[16] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proc. ACM conf. RecSys*, Barcelona, Spain. 2010, pp. 135-142.

[17] G. Zhao, and X. Qian, "Service objective evaluation via exploring social users' rating behaviors," 2015 *IEEE International Conference on Multimedia Big Data (BigMM)*. pp. 228-235.

[18] G. Zhao, X. Qian, and H. Feng, "Personalized recommendation by exploring social users' behaviors," in *Proc. MMM* 2014, pp.181-191.

[19] S. Jiang, X. Qian, J. Shen, Y. Fu, and T. Mei, "Author topic model-based collaborative filtering for personalized POI recommendations," *IEEE Trans. Multimedia*, 2015, vol.17, no.6, pp.907-918.

[20] Y. Ming, J. Sang, and C. Xu, "Unified YouTube video recommendation via cross-network collaboration," in *Proc. ICMR'15, ACM,* 2015, pp.19-26.

[21] J. Sang, T. Mei, and C. Xu, "Activity sensor: check-in usage mining for local recommendation," *ACM Transactions on Intelligent Systems and Technology (TIST)* , Volume 6, Issue 3, 2015.

[22] H. Feng, and X. Qian, "Recommendation via user's personality and social contextual," in *Proc. CIKM* 2013, ACM, pp.1521-1524.