Opinion-Driven Matrix Factorization for Rating Prediction

Štefan Pero and Tomáš Horváth

Institute of Computer Science,
Pavol Jozef Šafárik University, Košice, Slovakia
stefan.pero@student.upjs.sk, tomas.horvath@upjs.sk

Abstract. Rating prediction is a well-known recommendation task aiming to predict a user's rating for those items which were not rated yet by her. Predictions are computed from users' explicit feedback, i.e. their ratings provided on some items in the past. Another type of feedback are user reviews provided on items which implicitly express users' opinions on items. Recent studies indicate that opinions inferred from users' reviews on items are strong predictors of user's implicit feedback or even ratings and thus, should be utilized in computation. As far as we know, all the recent works on recommendation techniques utilizing opinions inferred from users' reviews are either focused on the item recommendation task or use only the opinion information, completely leaving users' ratings out of consideration. The approach proposed in this paper is filling this gap, providing a simple, personalized and scalable rating prediction framework utilizing both ratings provided by users and opinions inferred from their reviews. Experimental results provided on a dataset containing user ratings and reviews from the real-world Amazon Product Review Data show the effectiveness of the proposed framework.

Keywords: rating prediction, opinion mining, recommendation, personalization.

1 Introduction

Rating prediction is a well-known recommendation task [20] aiming to predict a user's ratings for those items which were not rated yet by the user. Very precise recommendations can be computed with matrix factorization techniques [11] by considering only the sparse user-item-rating relation on the input [17].

Rating is a coarse evaluation of user's opinion on items. Rating scales are often quite roughly grained (usually from 1 to 5 stars), forcing the user to choose either a lower or a higher rating in a case when her real attitude lies in between these two values. Moreover, user can also make a slip during the rating process.

Let's consider the following real-world examples from the data used in our experiments (described later in the section 6) of user ratings and reviews provided on items in the table 1: In the first row of this table, the attitude of user $u1$ on item $i2$ expressed in her review seems to fall between neutral and good (let's say

S. Carberry et al. (Eds.): UMAP 2013, LNCS 7899, pp. 1–13, 2013.

3.5 stars) but she had to choose either the neutral (3 stars) or the good (4 stars) rating. Probably, she has chosen the rating 4 because of her positive bias, i.e. a systematic tendency of her to give better ratings. In the next row of the table, however, the rating is probably a slip since the opinion of the same user on the item $i3$ expressed in her review is clearly good or excellent (4 or 5 stars) despite her poor (2 stars) rating given for that item. Moreover, while she rated the item $i2$ "above" her opinion expressed in the corresponding review, in the case of the item $i3$, the situation is the opposite.

Table 1. User ratings and reviews on items with the opinion (sentiment) scores computed from the reviews according to the algorithm presented in the section 4

User	Item	Rating	Review	Opinion score
u1	i2	4	"This CD is okay. Not a bad album at all. It just has to grow on you."	3
u1	i3	2	"This is a must have for a music DVD collection. So many great singers!"	4
u2	i1	5	"A good read. Not a bad beginner's guide. Some technology is a little dated, but useful."	4
u2	i4	5	"Interesting. A good overview, but Rumbelow's book is better researched and more comprehensive."	4
u3	i1	4	"Excellent. This book is an excellent novel. Excellent plot and characters. And riveting."	4
u3	i3	2	"Bad Vocals. Claude Williams is a good player, but this album is marred by some lousy vocal tracks."	3
u4	i3	1	"I didn't like this book and I couldn't get into it. The only thing I liked was the stripper with a Thumper tattoo. That was cute."	2
u4	i4	4	"Nice Good illustrations, tight bios of the buildings, not very gripping. For Main Line fans."	4

Biased matrix factorization techniques [11] computing the predictions from users' past ratings seem to be an adequate solution to the above mentioned issue of users' biases. However, real data shows that there is a variance in user's bias, i.e. a single user underrate some items while overrate some other items compared to her opinion expressed in her reviews (see the histograms of the differences between ratings and opinions in Figure 2).

[24,25] and also [6,13,18] claim that using only opinions (expressed in users' reviews) instead of ratings leads to better recommendation. Only [6,13] and [18] from these works are focused on the rating prediction problem. These works suggest that users' reviews are valuable sources of their opinions on items, often more accurate than their ratings and thus, do not consider rating information when learning the prediction model.

On the other hand, reviews are also biased by many factors such as the vocabulary of a user, sentences in a review not related to items or not expressing opinions, etc. In these cases, ratings can be used to infer some user-specific knowledge about reviews. Recommendation techniques utilizing both ratings and opinions were developed [1,12,10,22,4], however all of them are focused on the item recommendation task. Moreover, except [22], the mentioned approaches are either non-personalized, non-scalable or need more implementation effort.

In this paper, we focus on filling the above mentioned gaps by combining ideas from [13,18] (only opinions used) and [11] (only ratings used). The contributions of this paper are the following:

- Introducing a personalized, scalable and easy-to-implement framework for rating prediction utilizing opinions (sentiments) inferred from user reviews (section 4).
- An overview and analysis of the state-of-the-art recommendation techniques which utilize information derived from user reviews (section 5).
- Analysis of the relation between user ratings and derived opinions on a real-world Amazon Product Review Data (section 6).
- Experimental comparison of the proposed framework with some of the related approaches on the mentioned rel-world Amazon Data (section 7).

2 Matrix Factorization for Rating Prediction

Let $\mathcal{U}$ and $\mathcal{I}$ be the set of users and items, respectively, and $\mathcal{V} \subset \mathbb{R}$ be the set of values users can assign to items. A mapping $r : \mathcal{U} \times \mathcal{I} \rightarrow \mathcal{V}$ which defines a value r_{ui} assigned to an item i by the user u is called rating and is explicitly defined by a set of recorded past user-item feedbacks[1]

$$R = \{(u, i, r_{ui}) \mid u \in \mathcal{U},\ i \in \mathcal{I},\ r_{ui} \in \mathcal{V}\}$$

We usually split R into $R^{train}, R^{test} \subset R$ simulating users' past and future ratings, respectively, such that $R^{train} \cap R^{test} = \emptyset$.

The goal of rating prediction is, given R^{train}, to find a mapping $\hat{r} : \mathcal{U} \times \mathcal{I} \rightarrow \mathbb{R}$ such that

$$error(r, \hat{r}) = \sum_{(u,i,r_{ui}) \in R^{test}} (r_{ui} - \hat{r}_{ui})^2 \tag{1}$$

is minimal, where $\hat{r}_{ui}$ is the predicted rating given to the item i by the user u.

Biased matrix factorization [11] is a state-of-the-art approach to rating prediction, where R is viewed as a sparse matrix of type $\mathcal{V}^{|\mathcal{U}| \times |\mathcal{I}|}$ with r_{ui} being the values of its non-empty cells belonging either to R^{train} or R^{test} (similarly as mentioned above). The goal is to approximate a matrix R by the product of two smaller matrices $W \in \mathbb{R}^{|\mathcal{U}| \times k}$ and $H \in \mathbb{R}^{|\mathcal{I}| \times k}$ (where k is the number of factors), i.e. $R \approx WH^T$, and find vectors $b^{'} \in \mathbb{R}^{|\mathcal{U}|}$, $b^{''} \in \mathbb{R}^{|\mathcal{I}|}$ as well as a constant μ such that[2]

$$\sum_{r_{ui} \in R^{test}} (r_{ui} - \mu - b^{'}_u - b^{''}_i - w_u h_i^T)^2 \tag{2}$$

is minimal. The predicted rating $\hat{r}_{ui}$, given W, H, $b^{'}$, $b^{''}$ and μ is defined as

$$\hat{r}_{ui} = \mu + b^{'}_u + b^{''}_i + \sum_k w_{u_k} h_{i_k} \tag{3}$$

[1] Represented in the first three columns in the table 1.

[2] w_u (h_i) refers to the u-th (i-th) row of the matrix W (H), and, $b^{'}_u$ ($b^{''}_i$) refers to the u-th (i-th) element of the vector $b^{'}$ ($b^{''}$).

where μ refers to overall average rating and $b^{'}$, $b^{''}$ are the vectors of users' and items' biases, respectively, indicating how ratings of users or items deviate from μ (capturing systematic tendencies for some users to give higher ratings than others, and for some items to receive higher ratings than others).

The most popular factorization technique used in the recommender systems community, exploits stochastic gradient descent optimization [11] to find the parameters W and H by minimizing the following objective function

$$\sum_{r_{ui} \in R^{train}} (r_{ui} - \hat{r}_{ui})^2 + \lambda(\|W\|^2 + \|H\|^2 + b^{'2} + b^{''2}) \tag{4}$$

where $\hat{r}_{ui}$ is defined as in the equation 3 and λ is a regularization term to prevent the so-called over-fitting (i.e. when a model performs very well on the training data but poorly on the test data).

3 Opinion Mining

Opinion words are words that people use to express their positive or negative attitude to products or specific features of products. There are several techniques in the literature on opinion mining and sentiment analysis (see [15], for more details) from which we choose the most simple one:

Let the review, or comment, $c_{ui} = (w_1, \dots, w_n)$ of the user u on the item i be represented as a sequence of words. Each word in c_{ui} is either an opinion word or not. Opinion words are words that are primarily used to express subjective opinions. Clearly, this is related to the existing work on distinguishing sentences (subsequences of c_{ui}) used to express subjective opinions from sentences used to objectively describe some information [8]. We use adjectives and phrases as opinion words collected in the Sentiment (Opinion) Lexicon [15] containing a list of about 6800 English words expressing positive or negative sentiments. We will denote the sentiment lexicon as $\mathcal{S} = \mathcal{S}^+ \cup \mathcal{S}^-$ with $\mathcal{S}^+$ and $\mathcal{S}^-$ denoting its subsets of positive and negative sentiment words, respectively.

For each word $w_j \in c_{ui}$ we identify its semantic orientation $s(w_j)$ as

$$s(w_j) = \begin{cases} +1, \text{ if } & w_j \in \mathcal{S}^+ \\ -1, \text{ if } & w_j \in \mathcal{S}^- \\ 0, \text{ if } & w_j \notin \mathcal{S} \end{cases} \tag{5}$$

The overall orientation $o_{ui} \in [-1, +1]$ of the review c_{ui}, what we call the opinion or the sentiment of the review, is computed as

$$o_{ui} = \frac{\sum_{w_j \in c_{ui}} s(w_j)}{|\{w_j \in c_{ui} | w_j \in \mathcal{S}\}|} \tag{6}$$

There are plenty of other opinion mining and sentiment analysis techniques (see [16] for more details) as well as matrix factorization techniques (a unified view of which is introduced in [21]), which could be used in our framework.

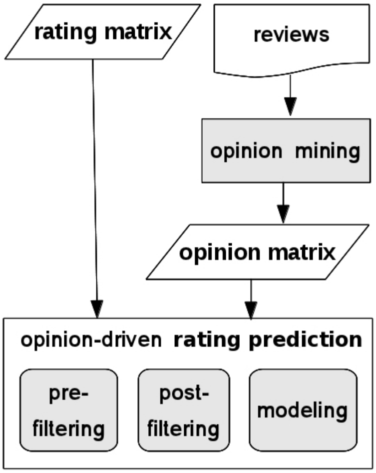

Fig. 1. The proposed framework

4 The Proposed Framework

Our framework is illustrated in Figure 1. Users' ratings (the rating matrix) and textual reviews on items are required on the input. The process consists of two steps, in general: First, an opinion matrix is created from the reviews by post-processing (discretizing) the result of the used opinion mining technique. In the second phase, a matrix factorization technique for rating prediction is employed utilizing this opinion matrix.

Similarly to the three paradigms of utilizing context in recommender systems [2], we distinguish three different approaches in the second step of our framework depending on which stages of the rating prediction process is the opinion matrix implied in. These are i) opinion pre-filtering, ii) opinion post-filtering and iii) opinion modeling. In all of these three approaches, we use the above presented biased matrix factorization.

However, even if we employ the three paradigms of context-aware recommendation, it is important to note that we do not perceive opinions as context here, but rather as regularizers of the prediction model.

4.1 The Opinion Matrix

As mentioned above in the section 3, the overall opinion expressed in the review of the user u on the item i is represented by a number $o_{ui} \in [-1, 1]$, in general, which expresses the polarity of her review on the given item with -1 and 1 being the most negative or positive, respectively. Ratings also expresses the polarity of the user's attitude to an item, with $min\,\mathcal{V}$ and $max\,\mathcal{V}$ being the most negative or positive, respectively. Since $\mathcal{V}$ usually consists of integers, discretizing the $[-1, 1]$ interval to $|\mathcal{V}|$ distinct values results in the same set of values for both opinions

o_{ui} and ratings r_{ui}. In our experiments we discretized the interval $[-1, 1]$ to $|\mathcal{V}|$ distinct values in an equidistant manner.

Similarly to ratings, discretized opinions can be viewed as a mapping $o : \mathcal{U} \times \mathcal{I} \to \mathcal{V}$ defined explicitly as[3]

$$O = \{(u, i, o_{ui}) \mid u \in \mathcal{U},\ i \in \mathcal{I},\ o_{ui} \in \mathcal{V},\ (u, i, r_{ui}) \in R\}$$

There is one assumption our framework is based on, namely that for all triple $(u, i, r_{ui}) \in R$ there is a corresponding triple $(u, i, o_{ui}) \in O$, i.e. users should provide both ratings and reviews for items.

As in the case of ratings (see the section 2), the opinion matrix O is considered as a sparse matrix of the type $\mathcal{V}^{|\mathcal{U}| \times |\mathcal{I}|}$, too, with o_{ui} being the values of its non-empty cells.

4.2 Opinion Pre-filtering

In this approach, opinions are used to pre-process the data for the recommendation technique being used.

The input relation matrix $\overline{R}^{train} \in \mathbb{R}^{|\mathcal{U}| \times |\mathcal{I}|}$ is created from the relations (matrices) R^{train} and O as

$$\overline{R}^{train} = \{\overline{r}_{ui} | \overline{r}_{ui} = \alpha r_{ui} + (1 - \alpha) o_{ui},\ r_{ui} \in R^{train},\ o_{ui} \in O\}$$

where $\alpha \in (0, 1)$. Thus, we modify the ratings in the train set such that they are closer to the opinions. This pre-processing step only affects the ratings in the training set, the ratings in the test set remain untouched.

Next, we factorize $\overline{R}^{train}$ to matrices $\overline{W} \in \mathbb{R}^{|\mathcal{U}| \times k}$ and $\overline{H} \in \mathbb{R}^{|\mathcal{I}| \times k}$ in a usual way as described in the section 2. The predicted rating $\hat{r}_{ui}$ of the user u for the item i is then defined according to the equation 3 as

$$\hat{r}_{ui} = \mu + b_u^{'} + b_i^{''} + \sum_k \overline{w}_{u_k} \overline{h}_{i_k} \tag{7}$$

4.3 Opinion Post-filtering

In this approach, we deal with two matrices R^{train} and the corresponding matrix O^{train} defined as

$$O^{train} = \{o_{ui} \in O | r_{ui} \in R^{train}\}$$

i.e. we keep only those cells from O which correspond to the cells of the train part of the rating matrix.

Then, R^{train} is factorized to matrices $W \in \mathbb{R}^{|\mathcal{U}| \times k}$ and $H \in \mathbb{R}^{|\mathcal{I}| \times k}$, while O^{train} is factorized to matrices $P \in \mathbb{R}^{|\mathcal{U}| \times l}$ and $Q \in \mathbb{R}^{|\mathcal{I}| \times l}$. In this way, we get two "interim" prediction models: $\hat{r}_{ui}^{'}$ for the ratings, defined exactly as $\hat{r}_{ui}$ in the equation 3, and

$$\hat{o}_{ui}^{'} = \mu^o + b_u^{'o} + b_i^{''o} \sum_l p_{u_l} q_{i_l} \tag{8}$$

[3] Represented in the first, second and last columns in the table 1.

for the opinions, where $\mu^o, b_u^{'o}, b_i^{''o}$ refer to overall sentiment (opinion) averages, and user and item sentiment (opinion) biases, respectively.

The predicted (post-filtered) rating $\hat{r}_{ui}$ of the user u for the item i is a linear combination of $\hat{r}^{'}$ and $\hat{o}^{'}$

$$\hat{r}_{ui} = \alpha\, \hat{r}^{'}_{ui} + (1 - \alpha)\, \hat{o}^{'}_{ui} \tag{9}$$

where $\alpha \in (0, 1)$.

4.4 Opinion Modeling

In the two previous approaches we used opinions explicitly in the pre- or the post-processing steps. Here, opinions are used implicitly in the factorization (or modeling) phase by changing the objective function (equation 4) to

$$\sum_{r_{ui} \in R^{train}} \alpha (r_{ui} - \hat{r}_{ui})^2 + \lambda (\|W\|^2 + \|H\|^2 + b^{'2} + b^{''2}) \tag{10}$$

where

$$\alpha = \begin{cases} \delta \in (0,1), & \text{if} \quad r_{ui} < \hat{r}_{ui} \leq o_{ui} \\ & \text{or} \quad o_{ui} \leq \hat{r}_{ui} < r_{ui} \\ 1, & \text{otherwise} \end{cases} \tag{11}$$

i.e. we just simply factorize R^{train} in a usual way (see section 2) but giving less weight to the prediction error if the predicted value $\hat{r}_{ui}$ lies between the rating r_{ui} and the opinion o_{ui} or it is equal to the opinion value.

5 Related Work

The earliest work [1] using consumer product reviews for recommendation is based on computing the qualities of the features of products from aggregating reviewers' opinions on these features weighted by the level of reviewers' expertise. User queries of the form "I would like to know if Sony W70 is a good camera, specifically its interface and battery consumption" are required at the input of the presented recommender system. A similar system is presented in [12]. Since both systems aggregate all opinions on a single item or on its features to one score, *neither provides personalized recommendations.*

The research hypothesis investigated in [6,13] and also in [18] is that sentiments of reviews are better indicators of users' attitude to items than coarse star ratings, and thus, *these works use only the reviews (leaving ratings out of consideration).* All these approaches consist of two main steps, the first of which is to infer for each review a numerical rating (an opinion) representing the sentiment level of the given review. The main difference from our approach lies in the second step, i.e. how the recommendations are made. In [6], domain and opinion specific meta-data are identified from reviews, and the final rating for a given item is predicted as an average rating of reviews sharing the same meta-data. In contrast to [6], a personalized recommendation approach is chosen in [13] and [18], where nearest-neighbor based collaborative filtering algorithms feeded by inferred opinions (sentiment scores) are used for recommendations.

A hybrid of content-based (CBF) and collaborative-filtering (CF) framework is presented in [10] where each item is represented as a vector consisting of the key aspects (relevant terms derived from user reviews and item descriptions) of items based on their importance values and sentiment scores. Such movie vectors (i.e. importance values and sentiment scores) are constructed for each user separately from the ratings and reviews of similar users to the given user. A binary ("recommendable" vs. "unrecommendable" item) classification model is learned from the derived aspect vectors for each user separately using classification techniques, which is then *used for item recommendation*.

The works introduced in [22,24,25] also *deal with item recommendation*. In [24], a nearest-Neighbor based collaborative filtering (CF) technique is used to recommend top-N items from a so-called virtual user rating matrix created from user reviews by sentiment classification. This matrix contains only binary values, i.e. a user likes or dislikes an item, regarding to the sentiment of her reviews on a given item. The presented framework is further augmented in [25] by considering also the keywords liked/disliked/unknown by the user. In a simple approach presented in [22], personalized recommendations (computed by a similarity-based CF technique) are further filtered according to the sentiments of experts' reviews on items. Thus, users are provided only with top-N items having positive reviews from experts.

A latent variable model for *content-based filtering* is introduced in [4], providing a supervised learning model for extraction of product features from reviews. The model can be trained on some available public datasets and then used to extract sentiments for reviews for which the rating is not provided. The presented model is implemented in the Opinion Space platform [3].

Similarly to our approach, the framework proposed in [19] assumes users to provide both ratings and reviews for items. However, *instead of a sentiment (opinion) score, a so-called "helpfulness" score of a review is considered*, derived from the feedbacks of other users provided on the given review (i.e. a ratio of users which found the given review helpful to all the users which have provided some feedback for the given review). Following the idea that a helpfulness score of a review indicates the quality of the corresponding rating, ratings are weighted by these helpfulness scores in the used factorization model.

The majority of the related work is focused on item recommendation. Works related to rating prediction are either non-scalable [6] or utilize only the opinions leaving ratings out of consideration [13,18], eventually, do not consider user sentiments (opinions) derived from reviews.

6 Data

We used the originally labeled Amazon Product Review Data[4] [9] in our experiments, which contains user-item-rating-review quadruples on movies, music or books. The rating scale is $\mathcal{V} = \{1, 2, 3, 4, 5\}$. We denote this dataset *original-data*. Since this dataset is very sparse (see the table 2) we created a smaller, much dense sample from it in the following way: We filtered out contributions from users who have provided fewer than 50 and more than 500 reviews. The resulting dataset is denoted as *sampled-data*. The main characteristics of these two datasets are shown in the table 2.

Table 2. Characteristics of datasets used in our experiments

dataset	#users	#items	#ratings = #reviews	sparsity (%)
original-data	2146275	1231018	5838898	0.000220994
sampled-data	4654	287666	606294	0.0452865

The histograms of differences between ratings and sentiment (opinion) scores inferred from reviews, for both datasets used are shown in Figure 2. Assuming that these differences are normally distributed we computed their mean and standard deviation: the mean is 0.44 and the standard deviation is 1.24 in case of original-data. In case of sampled-data, the mean of difference between ratings and opinions is 0.6 and the standard deviation is 1.53. This indicates that in general, users' ratings are a bit optimistic compared to their sentiments expressed in their reviews.

7 Experiments

We implemented the opinion mining technique described in the section 3 on our own. For biased matrix factorization, the algorithm from the MyMediaLite Recommender System Library [5] was used and modified if it was necessary, e.g. in case of the opinion-modeling approach (equation 10). For the computation, we used a computing cluster with 7 computing nodes with 24GB of RAM, each of which has 16 cores.

7.1 Baselines

The first baseline we used is the biased matrix factorization (*BiasedMF*) considering only the rating information [11], where the predicted ratings $\hat{r}_{ui}$ are computed according to the equation 3. Motivated by the two related works [13] and [18], we considered only the inferred opinions for learning $\hat{r}$ in our second baseline (*OpinionMF*). Here[5] we first factorized the opinion matrix and predict ratings as $\hat{r}_{ui} = \hat{o}^{'}_{ui}$, computed according to the equation 8.

[4] Bing Liu, `http://liu.cs.uic.edu/download/data/`

[5] Note, that it is the same as setting $\alpha = 0$ in case of our pre-filtering approach.

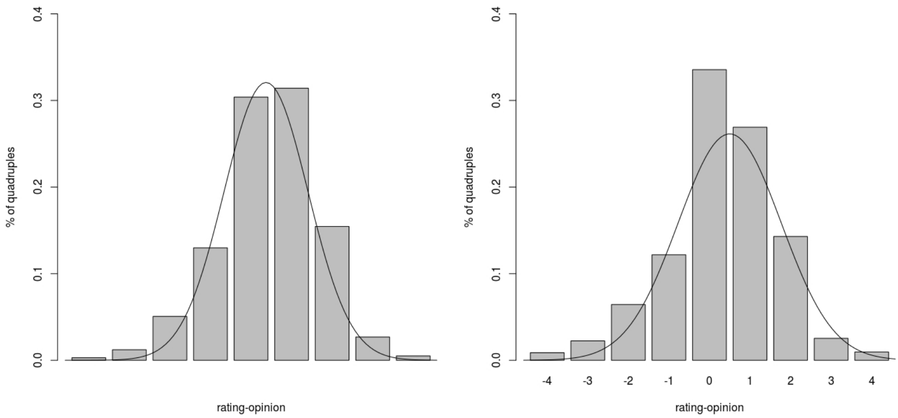

Fig. 2. The histogram of the ratios of user-item-rating-sentiment quadruples w.r.t. the differences in ratings and the computed sentiment (opinion) scores. On the left, histogram for sampled-data. On the right, histogram for original-data.

7.2 Hyper-parameters and Cross-Validation

We have used 5-fold cross-validation for testing the proposed framework as follows: In each of the 5 iterations, one fold was used for testing. From the remaining four folds, three were used for tuning the hyper-parameters of the model validated on the remaining fold. Hyper-parameters (number of factors, number of iterations, learn rate and regularization term) were tuned using grid search. The final model was trained with the best found hyper-parameter combination using all the remaining four folds. We set the value of the parameter α to 0.5.

7.3 Results

In the table 3, the average of the RMSE over the 5 folds is presented (for sampled-data as well as original-data) for the two baselines (denoted as *BiasedMF* and *OpinionMF*) and the proposed three approaches in our framework (denoted as *Pre*, *Post* and *Modeling*).

Table 3. RMSE averaged over the 5 folds

Dataset	BiasedMF	OpinionMF	Pre	Post	Modeling
sampled-data	0.9709	0.9486	0.9409	0.9060	0.9611
original-data	0.9712	0.9415	0.9542	0.9088	0.9645

The results on original-data are very similar to the results on sampled-data, even if the sparsity of these two datasets are considerably different.

The results of *BiasedMF* are significantly worse than the results of all the other approaches. Clearly, *Post* is the winner which provides significantly better results comparing to all other approaches the reason of what, in our opinion, could be that it behaves as a kind of an ensemble technique of two factorization models which tends to outperform single models [23]. *Pre* is significantly better than *BiasedMF* and *Modeling*, while *Modeling* is only significantly better than *BiasedMF*. Student's t-test was used to test statistical significance: except the case of *Modeling* vs. *BiasedMF* where the confidence level is 97%, all the other differences are significant with confidence level 99%.

The results also experimentally justified (with confidence level 97%) the ideas presented in [13] and [18], namely, that pure reviews tend to be better predictors than coarse ratings.

8 Conclusions

A generic matrix factorization framework for rating prediction utilizing user reviews was introduced in this paper. The proposed framework, based on biased matrix factorization, is a scalable, personalized and easy to implement recommendation framework. A simple opinion mining technique was used to derive users' sentiment on items. There is one assumption to the presented work, namely, that users provide both ratings and reviews for items.

We also provided a thorough review of the related state-of-the-art techniques providing with their main characteristics. The main idea of these works is that opinions expressed in users' reviews are good predictors of the ratings, and some works claim that opinions are even better predictors of ratings than are the ratings themselves. In this work, we deal with a combined usage of ratings and inferred sentiment (opinion) scores for rating prediction.

There are still some remaining issues to investigate regarding the proposed framework, e.g. how to deal with the cold-start problem, missing reviews, language biases present in reviews, and, how and in what extend does the proposed framework depend on the choice of the opinion mining algorithm. Following the ideas presented in [7] and [14], parallelization of our framework or its augmentation to include additional information (e.g. meta-data of users and items or additional relations between users), respectively, should be straightforward.

During the analysis of real-world data in our experiments, we have found out that there is a certain relationship between users' ratings on items and opinions expressed in their reviews on items. Since this paper was focused on rating prediction utilizing opinions, we would like to investigate this relationship between ratings and reviews from a user-modeling perspective in our future work.

Even if there are some remaining issues to investigate, experimental results show that the proposed framework is promising and worth of further research.

Acknowledgements. This work was partially supported by the research grants *VEGA 1/0832/12* and *VVGS-PF-2012-22*. We would like to thank to anonymous reviewers for their helpful and motivating comments and suggestions.

References

1. Aciar, S., Zhang, D., Simoff, S., Debenham, J.: Informed Recommender: Basing Recommendations on Consumer Product Reviews. IEEE Intelligent Systems 22, 3 (2007)
2. Adamovicius, G., Tuzhilin, A.: Context-Aware Recommender Systems. In: Ricci, F., et al. (eds.) Recommender Systems Handbook. Springer (2011) ISBN 978-0-387-85819-7
3. Faridani, S., Bitton, E., Ryokai, K., Goldberg, K.: Opinion space: a scalable tool for browsing online comments. In: Proceedings of the 28th International Conference on Human Factors in Computing Systems. ACM, New York (2010)
4. Faridani, S.: Using canonical correlation analysis for generalized sentiment analysis, product recommendation and search. In: Proceedings of the 5th ACM Conference on Recommender Systems. ACM, New York (2011)
5. Gantner, Z., Rendle, S., Freudenthaler, C., Schmidt-Thieme, L.: MyMediaLite: A Free Recommender System Library. In: Proceedings of the 5th ACM International Conference on Recommender Systems, Chicago, USA (2011)
6. Ganu, G., Elhadad, N., Marian, A.: Beyond the Stars: Improving Rating Predictions using Review Text Content. In: 12th International Workshop on the Web and Databases, Providence, Rhode Island, USA (2009)
7. Gemulla, R., Nijkamp, E., Haas, P.J., Sismanis, Y.: Large-scale matrix factorization with distributed stochastic gradient descent. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, New York (2011)
8. Hu, M., Liu, B.: Mining and Summarizing Customer Reviews. In: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Seattle, Washington, USA (2004)
9. Jindal, N., Liu, B.: Opinion Spam and Analysis. In: Proceedings of First ACM International Conference on Web Search and Data Mining (WSDM-2008), Stanford University, Stanford, California, USA (2008)
10. Ko, M., Kim, H.W., Yi, M.Y., Song, J., Liu, Y.: Movie Commenter: Aspect-based collaborative filtering by utilizing user comments. In: 7th International Conference on Collaborative Computing, Orlando, FL, USA (2011)
11. Koren, Y., Bell, R., Volinsky, C.: Matrix Factorization Techniques for Recommender Systems. Computer 42(8) (2009)
12. Kuroiwa, T., Bhalla, S.: Aizu-BUS: need-based book recommendation using web reviews and web services. In: Bhalla, S. (ed.) DNIS 2007. LNCS, vol. 4777, pp. 297–308. Springer, Heidelberg (2007)
13. Leung, C.W.K., Chan, S.C.F., Chung, F.: Integrating collaborative filtering and sentiment analysis: A rating inference approach. In: Proceedings of the ECAI 2006 Workshop on Recommender Systems (2006)
14. Lippert, C., Weber, S.H., Huang, Y., Tresp, V., Schubert, M., Kriegel, H.-P.: Relation Prediction in Multi-Relational Domains using Matrix Factorization. In: NIPS Workshop on Structured Input Structure Output (2008)
15. Liu, B.: Sentiment Analysis and Subjectivity. In: Handbook of Natural Language Processing, 2nd edn. (2010)
16. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval 2(1-2) (2008)
17. Pilászy, I., Tikk, D.: Recommending new movies: even a few ratings are more valuable than metadata. In: Proceedings of the Third ACM Conference on Recommender Systems. ACM, New York (2009)

18. Poirier, D., Fessant, F., Tellier, I.: Reducing the Cold-Start Problem in Content Recommendation through Opinion Classification. In: Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology. IEEE Computer Society, Washington, DC (2010)
19. Raghavan, S., Gunasekar, S., Ghosh, J.: Review quality aware collaborative filtering. In: Proceedings of the Sixth ACM Conference on Recommender Systems (RecSys 2012). ACM, New York (2012)
20. Ricci, F., Rokach, L., Shapira, B., Kantor, P.B.: Recommender Systems Handbook. Springer (2011)
21. Singh, A.P., Gordon, G.J.: A Unified View of Matrix Factorization Models. In: Daelemans, W., Goethals, B., Morik, K. (eds.) ECML PKDD 2008, Part II. LNCS (LNAI), vol. 5212, pp. 358–373. Springer, Heidelberg (2008)
22. Singh, V.K., Mukherjee, M., Mehta, G.K.: Combining collaborative filtering and sentiment classification for improved movie recommendations. In: Sombattheera, C., Agarwal, A., Udgata, S.K., Lavangnananda, K. (eds.) MIWAI 2011. LNCS, vol. 7080, pp. 38–50. Springer, Heidelberg (2011)
23. Wu, M.: Collaborative filtering via ensembles of matrix factorizations. In: KDD Cup and Workshop at the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Jose, CA, USA (2007)
24. Zhang, W., Ding, G., Chen, L., Li, C.: Augmenting Chinese Online Video Recommendations by Using Virtual Ratings Predicted by Review Sentiment Classification. In: Proceedings of the 2010 IEEE ICDM Workshops. IEEE Computer Society, Washington, DC (2010)
25. Zhang, W., Ding, G., Chen, L., Li, C.: Augmenting Online Video Recommendations by Fusing Review Sentiment Classification. In: Proceedings of the 2nd ACM RecSys 2010 Workshop on Recommender Systems and the Social Web. ACM (2010)