Improving predictions with user and item profiling

Maria Laura Clemente

CRS4, Center of Advanced Studies, Research and Development in Sardinia, Italy

clem@crs4.it

Abstract. The value of Biased Matrix Factorization algorithms in recommender systems, based only on numeric ratings, has already been demonstrated. Improvements in predictions can be achieved adding more information, for example considering user generated textual reviews, although the lack of rules increases the level of difficulty in machine learning methodologies. The aim of the presented activity is to experiment the online Latent Dirichlet allocation to build user and item profiles in order to improve predictions obtained with a Biased Matrix Factorization algorithm. For the experimental analysis the Yelp data set was used, limited to the Restaurant category. Applying a 5-fold cross validation promising results were obtained in terms of Root Mean Squared Error (RMSE).

Keywords: user profiling; collaborative filtering; social networks; matrix factorization; online Latent Dirichlet Allocation.

1 Introduction

A key element of the research activities related to user profiling for recommender systems is the dataset. From the features and the context of the dataset depend also the results. Some algorithms which give poor results with a dataset can produce better predictions if used with different data [1]. This consideration is particularly relevant when the results of an experimental activity are used for the implementation of a real context of application.

The input of machine learning methodologies typically applied in user profiling can be substantially classified in two main types: the more simple one consists in a single numeric rating (such as the number of stars, from one to five); the other one is made of textual reviews written by the users. Each of these two types of datasets has been used as input of different families of algorithms for user profiling and recommender systems. Generally the numeric ratings (which form the *matrix of the ratings*, having a row for each user and a column for each item) are a valid input for collaborative filtering algorithms such as the item-based, or more recently, the biased matrix factorization. Many datasets provide also more information about the users (such as age, gender, occupation, etc.) and the items (for example, if the items are restaurants, address, city, and categories). Depending on the context of application, these information can be used to improving the results; for example, in case of movies, during the well-known Netflix competition, a great improvement was achieved in predictions by considering the time of release in a neighborhood-based methodology [2]. Sometimes the numeric ratings are more than one, and each rate is related to pre-defined topics (for example, in the case of Restaurants the topics can be: service, quality of food, etc.). These rates can be summarized to one rate only in order to be analyzed by the same family of algorithms used to work with the *matrix of the ratings*.

The input made of textual reviews is suitable for a completely different type of algorithms, which analyze the text in order to understand for example the topics (such as the *online Latent Dirichlet Allocation*, here after *online LDA* [3]) or the positive, negative or neutral meaning (opinion mining methodologies [7]). The research in user profiling based on textual reviews is particularly challenging for the lack of rules, because any word is allowed and slang, exclamations, emoticons, and also misspellings are accepted. For this reason user generated textual reviews are more difficult to be analyzed, but at the same time they can be effectively used to try to better understand the users' preferences.

The datasets providing both types of input, the numeric ratings and the textual reviews, have a special value because they provide the researchers with more complete information allowing to explore new strategies of knowledge discovery. There are some dataset available on the internet of this family commonly used for research purposes, such as *yelp* and *amazon*.

The presented activity explores a new way to improve the predictions coming from a Biased Matrix Factorization algorithm [4], with a particular use of online LDA methodologies [3] and it has been tested with the *yelp* dataset provided for the RecSys 2013 competition (other versions are available as well) limited to the Restaurant category, which was the most represented.

The rest of the article is structured as follows: the second section is about related works; the third section presents the methodology of the experimental activity; the fourth section is dedicated to the experiments, providing more details about the research activity and how it has been carried out; the fifth section shows results and explains them; lastly the conclusions sum up what has been achieved with the presented activity and how it could be further developed.

2 Related works

The efficacy of Biased Matrix Factorization algorithms in making predictions for recommender systems has been demonstrated in many works [4], and the improvements of predictions generated with this methodology considering other information is an interesting activity.

Online Latent Dirichlet Allocation (online LDA) is able to deal with the issues of human generated textual reviews and is extremely fast and effective in topic discovery [3].

In machine learning the usage of combinations of different approaches to analyze user generated textual reviews and numeric ratings is very common. In general the analysis of the two kinds of input is dealt with independent methodologies: Natural Language Processing to analyze textual reviews, and collaborative filtering to work on the numeric ratings, the output are then merged through ensemble methodologies [5].

The Yelp dataset have been already used for research activities in previous works [6][7][8][9][10]. Particularly relevant is the work by McAuley and Leskovec [6] which combines the latent features coming from a matrix factorization algorithm with the latent review topics obtained with LDA.

In the presented activity, the same algorithms are applied but in a different way, as explained here after. The Biased Matrix Factorization was used to make predictions of the star ratings which the users in the test set could assign to the restaurants. From this starting point, the predictions were corrected with a targeted use of online LDA methodologies. The topic discovery, which was output of the online LDA methodology, was used to build arrays of user preferences and restaurant preferences, working as user profiles and restaurant profiles, in order to find a level of affinity between each couple of user and restaurant in the set of the reviews. To train and test the methodology, the Yelp dataset was split into 5 different groups in order to follow a 5-fold cross validation and the Root Mean Squared Error (RMSE) was used to evaluate the results. The combination of the two algorithms, as more deeply explained in the rest of the article, produced a slight improvement in terms of RMSE, which encourages to further exploration in future work.

3 Experimental design and Methodology

The presented approach explores a new way to consider numeric ratings and textual reviews to improve predictions. In particular, the numeric ratings were used as input of a Biased Matrix Factorization algorithm which produced an output in terms of predictions; the predictions were then refined according to user and restaurant profiles built using the online LDA on the textual reviews. A baseline algorithm made of average values was used in order to understand which results could be achieved without involving any time consuming computation; the results obtained with the Biased Matrix Factorization can be considered as a further baseline, because the hypotheses to be tested is to try to improve them.

To perform the experimental activity, the Yelp dataset was firstly divided into five different groups used in the cross-validation during the whole experimental activity.

For simplicity, all the results are expressed in terms of RMSE, which is very common in the field of the algorithm for numeric predictions.

The steps of the methodology are explained in more details in the following subsections.

3.1 The evaluation criteria

To evaluate the results, the Root Mean Squared Error (RMSE) was used which is based on the error given by the difference between each prediction P and the actual numeric rating r for all the N reviews of the test set:

$$RMSE = \sqrt[2]{\frac{\sum_{i=1}^{N}(\mathrm{P}_i - r_i)^2}{N}} \quad (1)$$

3.2 The Yelp dataset

The research activity was carried out using all the 126744 textual reviews and star ratings (from 1 to 5) related to the category *Restaurant* of the Yelp dataset provided for the RecSys2013 (which is available at www.kaggle.com/c/yelp-recsys-2013/leaderboard). The train set of the Yelp dataset was divided into 5 groups of 31686 for cross-fold validation.

3.3 The algorithm made of averages (AVGs)

The baseline algorithm, made of averages, had the only purpose to be used as landmark to help in defining the lowest acceptable level of RMSE.

More in detail, each predicted rate given by a user to a restaurant in the test set, is calculated as a weighted average between the average of all the rates given by the user in the training set, and the average of all the rates received by the restaurant in the training set. For all the cold start couples of users and restaurants in the test set which were not present in the training set, the global average of the training set was used instead.

3.4 The Biased Matrix Factorization (BMF)

The BMF [4] algorithm used to make the predictions is already known to be very effective for recommender systems based only on the numeric ratings given by the users to the items (in this case to the restaurants). For the experimental activity the apache Mahout Taste library [11] [12] was used, in particular the learning algorithm was the Stocastic Gradient Descent Factorizer (a valid implementation of the algorithm explained in [4] and [13]), while the predictions were made calling the Singular Value Decomposition working as Recommender. The algorithm is already well known so here after only a short description will be given. Figure 1 shows a scheme of the factorization, which starting from the sparse matrix R of the numeric ratings (given by M users to N restaurants), allows to build the two matrix P and QT of latent factors (related to users and restaurants) which make it possible to predict all the unknown ratings. Starting from a fixed value of factors K, the matrix P is MxK, while matrix Q is NxK.

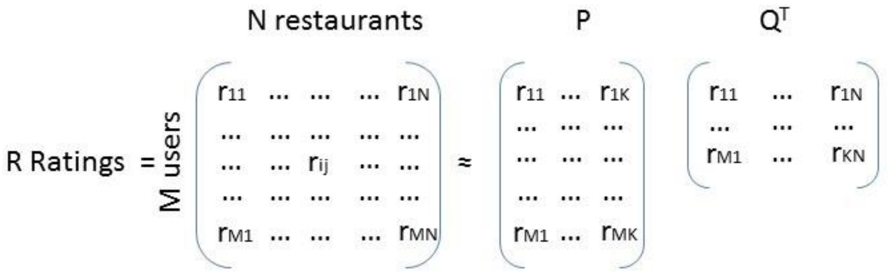

Fig. 1. The schema of Matrix Factorization

The algorithm is called *biased* because it takes into account the component of the ratings depending on the different ways users give ratings to the restaurants (due to their personalities, a person may give higher ratings than another one). When a user or a restaurant was missing in the training set, the prediction was made based on average values (as in the baseline algorithm).

3.5 The online Latent Dirichlet Allocation (online LDA)

In the experimental activity the online LDA [3] was applied using the libraries *nltk* [14] and *gensim* [15][16][17] for topic discovery, based on word frequency in the textual reviews. In particular, to each of the five groups in the dataset, after the tokenization (through the *nltk.tokenize.RegexpTokenize*), filtering out the English stop-words (using the *stop_words* package) and found stem words, the five dictionaries and the five corpora were saved. Dictionary and corpora were the input for the LDA model generation, which could take place very quickly thanks to the *ldamulticore* package of *gensim*. At this point the LDA model was generated many times in order to find a reasonable number of topics each time (10, 20, 25, 30, 50, and 60); every topic is represented by a list of words and the probability to find them in sentences related to that particular topic. In [18] a research activity was carried out using the same dataset using 50 topics. After the LDA model generation, the topics for each train set could be analyzed and this was really an interesting part of the activity which deserves some more explanation.

For example in the case of 10 topics, words belonging to distinct topics, such as Mexican and Sushi, resulted mistakenly together, suggesting that the best number of topics must be greater than 10:

```
(0.021*taco + 0.020*sushi + 0.017*good + 0.016*mexican +
0.014*salsa + 0.013*roll + 0.012*chip + 0.011*burrito +
0.010*like + 0.010*bean + 0.009*order + 0.009*fish +
0.009*food + 0.007*tortilla + 0.007*place')
```

It can be noticed that there are other words, such as *food*, *good*, and *order*, which are present, but they can be considered less characterizing, because they are very common words talking about restaurants and for this reason they have not been considered in the following of the paper.

The same words can be found in the case of 20 topics, this time separated, as they should be, and with higher values of probability, but in the *sushi* group it is possible to see the intrusion of words belonging to other groups, such as *thai*:

```
(0.039*taco + 0.030*mexican + 0.026*salsa + 0.020*chip +
0.020*burrito + 0.018*bean + 0.014*food + 0.013*good +
0.013*tortilla + 0.010*margarita + 0.010*order +
0.010*chees + 0.010*rice + 0.009*enchilada + 0.008*carn')
(0.100*sushi + 0.081*roll + 0.062*thai + 0.020*fish +
0.016*tuna + 0.015*spici + 0.013*pad + 0.013*fresh +
0.012*japanes + 0.010*happi + 0.009*tempura +
0.009*sashimi + 0.009*best + 0.009*chef + 0.009*sake')
```

Both Mexican and Sushi topics are still two different groups in the case of 30 topics, with probabilities higher than the ones obtained with 20 topics:

```
(0.044*taco + 0.034*mexican + 0.029*salsa + 0.023*chip +
0.022*burrito + 0.020*bean + 0.016*good + 0.014*food +
0.014*tortilla + 0.011*chees + 0.010*order +
0.010*enchilada + 0.010*rice + 0.009*green + 0.009*carn')
(0.107*sushi + 0.083*roll + 0.036*fish + 0.022*tuna +
0.014*fresh + 0.014*japanes + 0.012*salmon + 0.011*spici
+ 0.010*tempura + 0.009*sashimi + 0.009*sake +
0.009*order + 0.008*good + 0.008*chef + 0.008*bar')
```

In the case of 50 Topics the mexican group has been divided in two:

```
(0.090*taco + 0.035*burrito + 0.033*salsa + 0.022*chip +
0.021*carn + 0.019*asada + 0.018*guacamol +
0.017*tortilla + 0.017*margarita + 0.013*mexican +
0.013*chipotl + 0.012*good + 0.011*fresh + 0.010*order +
0.010*fish')
(0.058*mexican + 0.041*bean + 0.034*food + 0.030*salsa +
0.026*chip + 0.023*green + 0.021*enchilada + 0.020*rice +
0.018*chile + 0.018*good + 0.017*chees + 0.016*chicken +
0.015*chili + 0.014*tortilla + 0.013*burrito')
(0.152*sushi + 0.123*roll + 0.021*tuna + 0.019*japanes +
0.017*fish + 0.017*fresh + 0.014*tempura + 0.013*spici +
0.013*sake + 0.013*chef + 0.012*sashimi + 0.012*salmon +
0.011*bar + 0.010*miso + 0.008*ra')
```

In a further analysis it would be interesting to explore why certain words related to Mexican food go to one of these two groups, while others go to the other one.

In the case of 60 Topics, the second group of the Mexican category has words related to the yelp web site, and the values of probability are decreasing:

```
(0.054*taco + 0.039*mexican + 0.036*salsa + 0.028*chip +
0.027*burrito + 0.024*bean + 0.018*tortilla + 0.016*food
+ 0.015*good + 0.012*enchilada + 0.012*carn + 0.012*chees
+ 0.011*order + 0.011*asada + 0.011*rice')
(0.091*margarita + 0.050*com + 0.041*yelp + 0.030*http +
0.026*tequila + 0.026*www + 0.023*quesadilla +
0.020*select + 0.018*mexican + 0.014*barrio + 0.013*baja
+ 0.013*biz_photo + 0.012*mmm + 0.011*chimi +
0.011*jerk')
(0.134*sushi + 0.107*roll + 0.025*fish + 0.023*tuna +
0.018*japanes + 0.017*fresh + 0.013*salmon + 0.013*spici
+ 0.012*sake + 0.012*tempura + 0.011*sashimi + 0.010*chef
+ 0.010*bar + 0.009*miso + 0.009*order')
```

This behavior seems to suggest that the optimum number of topics is lower than 60.

Once all the models were generated (one for each group and for each number of topics), for each user of the train set a profile was calculated depending on the occurrences of each topic in all the reviews made by him/her. In a similar way also the restaurants' profiles were built and saved. These profiles have the following meaning: user profiles list which aspects are really of interest for each user; while the restaurants profiles put in evidence the features characterizing them. The files of users' profiles and restaurants' profiles built by using the training set, were used to correct the values of predictions for the test set, according to the following criteria: when a prediction was related to a couple 'user,restaurant' having many topics in common in their profiles, the BMF prediction was increased multiplying it by a coefficient greater than 1. On the contrary, when the affinity between the user and the restaurant was very low, the BMF prediction was decreased multiplying it by a coefficient minor than 1. In all the other cases the BMF prediction was not changed. The values of the coefficients were chosen through the cross-fold validation.

3.6 The combination of the algorithms

In this section the combination of the BMF and online LDA algorithms is explained more in detail. Once the Yelp training set related to the Restaurant category was split into 5 different groups, for each training set, the numeric ratings were used as input for the BMF algorithm, while the textual reviews were the input for the online LDA methodology. As explained in 4.5, the output of the BMF Factorization is made of the two matrices of the latent factors: one related to the users and the other related to the restaurants. Then the prediction takes place, for all the couples (user, restaurant) of the test set. For each review of the test set, the prediction obtained with the BMF was corrected according to the level of affinity between the user and the restaurant.

In Figure 1 a scheme of the whole methodology is shown.

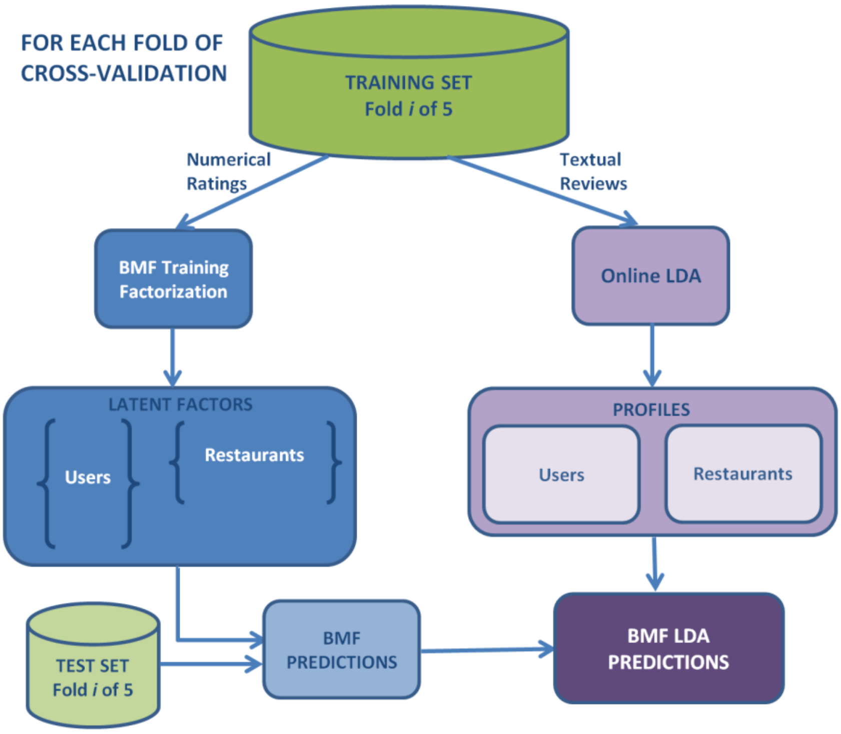

Fig. 2. Overall schema of the methodology

4 Results and discussion

The baseline algorithm, based on the weighted average between the user ratings and the restaurant ratings, produced a RMSE of 1.1139. An evaluation of predictions made by averages is always done in order to understand if the involvement of time consuming computation is worthy or not.

A better RMSE of 1.09535 could be easily achieved by a BMF, which is well known to be the most effective when used alone and the input consists of the matrix of the ratings [4]. It must be highlighted that for all the cases of cold start, for example when a user (or a restaurant) in the test set was not present in the training set, the value obtained with the baseline algorithm was used instead.

Some of the BMF predictions were out of range (greater than 5.0); setting these values equal to 5.0 allowed to have a RMSE of 1.09529.

The online LDA was run for topic discovery and to build users/restaurants profiles beforehand; then it was possible to calculate a level of affinity between users and restaurants. After the preparation of the profiles of users and restaurants, for each review in the test set, the level of affinity was obtained as the number of topics present in both the user profile and the restaurant profile. According to this level of affinity between the user and the restaurant of a review in the test set, the value of prediction already obtained with the BMF was corrected: incremented when the affinity of the user and the restaurant of a prediction was very high, decremented when the affinity was very low, and kept unchanged in all the other cases. With this approach it was possible to obtain a RMSE of 1.094605, which was the lowest and then the best value for the presented activity.

Table 1 shows which methodologies were used in each step of the experimental activity along with the overall best values of RMSE obtained.

Table 1. Summary of methodologies involved and related results in terms of RMSE

AVGs	BMF	Online LDA	RMSE
Yes			1.1139
Yes	Yes		1.09535
Yes	Yes	Yes	1.094605

It must be specified that these values are averages of all the 5 folds involved in the cross-validation. Actually for some folds it was possible to achieve lower RMSE values. For example the best value of 1.0902 was obtained for fold 2, which started from a RMSE of 1.0909 with BMF unchanged predictions. At the opposite was fold 3, which started from a RMSE of 1.1036 and improved to a lower RMSE of 1.1027 but was always particularly greater (worse) than the others. In a future work it would be interesting to further analyze the differences obtained between these two folds.

It is worth noting that this result was obtained without using any ensemble methodologies, but simply targeting the correction of predictions in the cases of great or no affinity.

Through an experimental activity which considered many criteria of correction and coefficients, the overall best RMSE were obtained with the following criteria of corrections:

```
if affinity < 10, BMF prediction * 0.98
if affinity > 400, BMF prediction * 1.0028
```

Table 2. Summary of the results obtained with different number of topics

Topics	RMSE
10	1.09467
20	1.09461
25	1.09558
30	1.09592
50	1.09471
60	1.09465

Although the additional use of online LDA can seem not significant because it produced a very small improvement in the RMSE obtained by BMF only, this result can be considered encouraging for further research activity, for example to deal with the cases of very sparse matrices of ratings and for the cold start problem.

5 Conclusions and future work

An experimental activity about predictions of star ratings was presented. For this kind of task it is already known that the Biased Matrix Factorization is the most effective algorithm to make predictions, if working alone (ensembles of more algorithms are able to produce better results). Social Networks often produce numeric ratings along with textual reviews. Since Biased Matrix Factorization does not work on textual reviews, the results obtained with BMF were improved analyzing the user and restaurant profiles calculated through the online LDA. This methodology brought to a slight improvement in terms of RMSE, but is promising for further exploration in future work, involving other datasets.

A further research activity could be carried out in order to find a correlation between the latent factors of users and restaurants provided by factorization, with the profiles of users and restaurants built using the output of the online LDA methodology.

6 Aknowledgement

This work was supported in part by the Italian Ministry of Education, University and Research through Project CagliariPort2020 under Grant SCN_00281.

7 References

1. Adomavicius, G.; Kwon, Y.O.; Zhang, J. Impact of Data Characteristics on Recommender Systems Performance. ACM Transactions on Management Information Systems (TMIS), vol. 3, issue 1. (2012).
2. Bell, R.M; Koren, Y. Improved Neighborhood-based Collaborative Filtering. Proceedings of KDDCup and Workshop, (2007).
3. Hoffman, M.; Blei, D.; Bach, F. Online Learning for Latent Dirichlet Allocation. In *Neural Information Processing Systems (NIPS)* (2010). Available online: https://www.cs.princeton.edu/~blei/papers/HoffmanBleiBach2010b.pdf
4. Koren, Y.; Bell, R.; Volinsky, C. Matrix factorization techniques for recommender systems. Computer 42 (8), 42-49, IEEE Computer Society (2009).

5. Jahrer, M.; Tosher, A.; Legentstein, R. Combining Predictions for Accurate Recommender Systems, Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 693-702, ACM, (2010).
6. McAuley, J.; Leskovec, J. Hidden Factors and Hidden Topics: Understanding Rating Dimensions with Review Text, RecSys '13 Proceedings of the 7th ACM conference on Recommender systems, 165-172, ACM, (2013).
7. Angioni, M.; Clemente, M.L.; Tuveri, F. Combining Opinion Mining with Collaborative Filtering, WEBIST 2015, 11th International Conference on Web Information Systems and Technologies (2015).
8. Ganu, G. ; Elhadad, N.; Marian, A. Beyond the stars: improving rating predictions using review text content, 12th International Workshop on the Web and Databases (WebDB), (2009).
9. Govindarajan, M. Sentiment Analysis of Restaurant Reviews Using Hybrid Classification Method, Proceedings of 2nd IRF International Conference, Chennai India (2014).
10. Trevisiol, M.; Chiarandini, L.; Baeza-Yates, R. Buon Appetito – Recommending Personalized menus (2014).
11. Owen, S.; Anil, R.; Dunning, T.; Friedman, E. Mahout in Action. Manning Publications Co., Shelter Island, ISBN 9781935182689 (2011)
12. Shelter, S.; Owen, S. Collaborative Filtering with Apache Mahout. RecSys Challenge (2012).
13. Tosher, A.; Jahrer, M.; Bell, R.M. The BigChaos solution to the Netflix grand prize, Netflix Prize Documentation, (2009).
14. Natural Language Toolkit, NLTK http://www.nltk.org/.
15. Gensim: models.ldamodel – Latent Dirichlet Allocation https://radimrehurek.com/gensim/models/ldamodel.html.
16. Jordan Barber, Latent Dirichlet Allocation (LDA) with Python https://rstudio-pubs-static.s3.amazonaws.com/79360_850b2a69980c4488b1db95987a24867a.html .
17. Sandulescu, V. Predicting what user reviews are about with LDA and gensim . http://www.vladsandulescu.com/topic-prediction-lda-user-reviews/
18. Huang, J.; Rogers, S.; Joo, E. Improving Restaurants by Extracting Subtopics from Yelp Reviews. iConference (2014).