LitRec vs. Movielens
A Comparative Study

Paula Cristina Vaz[1,4], Ricardo Ribeiro[2,4] and David Martins de Matos[3,4]
[1]*IST/UAL, Rua Alves Redol, 9, Lisboa, Portugal*
[2]*ISCTE-IUL, Rua Alves Redol, 9, Lisboa, Portugal*
[3]*IST, Rua Alves Redol, 9, Lisboa, Portugal*
[4]*INESC-ID, Rua Alves Redol, 9, Lisboa, Portugal*
{*paula.vaz, ricardo.ribeiro, david.matos*}*@inesc-id.pt*

Keywords: Recommendation Systems, Data Set, Book Recommendation.

Abstract: Recommendation is an important research area that relies on the availability and quality of the data sets in order to make progress. This paper presents a comparative study between Movielens, a movie recommendation data set that has been extensively used by the recommendation system research community, and LitRec, a newly created data set for content literary book recommendation, in a collaborative filtering set-up. Experiments have shown that when the number of ratings of Movielens is reduced to the level of LitRec, collaborative filtering results degrade and the use of content in hybrid approaches becomes important.

1 INTRODUCTION

The number of books, movies, and music items published every year is increasing far more quickly than is our ability to process it. The Internet has shortened the distance between items and the common user, making items available to everyone with an Internet connection. Recommendation systems emerged as an independent research area in the mid-1990's to help users find items that met their interests.

Recommendation systems performance relies on two factors: (a) algorithm performance and (b) data set availability. Algorithms can be easily implemented, but they need data sets to be tested, otherwise researchers cannot assess their accuracy in predicting user interests. Good data sets are hard to gather and take several days work to organize.

In this paper we present a comparative study between LitRec and Movielens to assess LitRec's suitability for book recommendation studies. Movielens (http://www.grouplens.org) is a movie recommendation data set collected by GroupLens (http://movielens.umn.edu) and has been used with success in recommendation systems research. LitRec is a new data set for literary book recommendation and combines *Project Gutenberg* (http://www.gutenberg.org) books with *Goodreads* (http://www.goodreads.com) ratings.

In this paper the authors aim to (a) present LitRec and (b) assess its suitability in recommendation studies.

The paper is structured as follows: Section 2 describes related work on recommendation systems and existing data sets. In Section 3 we describe the collaborative filtering (CF) algorithm and prediction equation, as well as different item representation. Section 4 describes the data sets, the evaluation metric, and the experiments. Finally, Section 5 concludes and points to future directions.

2 RELATED WORK

Recently, several data sets have become available to the recommendation systems research community. Of these, we highlight Movielens, a movie recommendation data set that has been extensively used. Movielens includes movie title, genre, ratings on a 1-5 star scale, and time stamp. Movielens also provides cross-validation. Book-Crossing data set (Ziegler et al., 2005) is a book recommendation data set with some limitations. It does not contain rating time stamp and books are not categorized. Moreover, Book-Crossing includes titles of the same book in different languages, e.g., "The Lord of the Rings" appears in English and Spanish as two different books, augmenting sparsity. Other data sets exist like Last.fm data set

for music recommendation (Celma, 2010) and Jester data set for joke recommendation (Goldberg et al., 2001), among others.

3 PROPOSED METHOD

We implemented an item-based CF algorithm using the k nearest neighbor (kNN) approach. kNN technique has been extensively used in recommendation research. Our algorithm registers the items that were preferred together in a $item \times item$ co-occurrence matrix, then, calculates similarity between the items that co-occur using the cosine similarity.

Predictions on item i for the user u are generated, first selecting the kNN and then, using the weighted sum in Equation 1 (where $cos(i, j)$ is the cosine similarity between items i and j, $r_{u,i}$ denotes the rating of user u on item i, N_u is the number of items in user u profile, and k is the number of items similar to item i. Item i is an item not preferred by user u).

$$P_{u,i} = \frac{1}{N_u} \sum_{j}^{k} cos(i, j) * r_{u,i} \quad (1)$$

We used three different item representations: (a) user vectors (b) Latent Dirichelet Allocation (LDA) (Blei et al., 2003) item topic vectors, and (c) topic vector using Latent Semantic Analysis (LSA) (Bellegarda and Juang, 2006). LDA and LSA were used for vector dimensionality reduction.

User vectors are obtained directly from the $item \times user$ rating matrix. Item topic vectors are obtained by applying LDA to the $item \times item$ co-occurrence matrix. LDA is typically used to represent documents as word topics applying a probabilistic approach. To define topics, it requires a $document \times word$ frequency matrix where each cell of the matrix contains the number of times a word appears in a document. We first tried to apply LDA to the $item \times user$ rating matrix, but using items as documents, users as words, and ratings as frequency. However, the results were not encouraging due to matrix sparsity and rating range (1-5). Therefore, we decided to apply LDA to the $item \times item$ co-occureence matrix and represent items as topics of items.

Finally, LSA topic vectors are obtained by applying singular value decomposition (SVD) to the $user \times item$ matrix. The resulting $item \times topic$ (V) matrix is then cut, using the standard cut-off value of 300 topics, and weighted by matrix Σ values. The cut-off value is the number of vector dimensions (topics) that will be used to represent items. This value depends on the data set, but empirical studies indicate typical cut-off values varying between 300 and 500 topics (Bellegarda and Juang, 2006).

4 EXPERIMENTS

In this section we describe LitRec data set, the evaluation metric, and the performed experiments.

4.1 Experimental Data

Movielens is a well known data set and it has been used in many recommendation systems studies. LitRec is a newly created data set. It combines documents from *Project Gutenberg* and ratings from *Goodreads*. In order, to make both sets comparable, we selected the 943 users with more ratings from LitRec, because Movielens has 943 users. Table 1 shows Movielens and LitRec parameters.

Table 1: Data set parameters.

	Movielens	LitRec
Items	1,682	2,598
Ratings	100,000	16,042
Sparsity	0.941	0.993
Ratings/user	106.04	17.01
Ratings/item	59.45	6.17

For each book, LitRec includes the author, rating date, added date, and read date. The rating and added date are set by *Goodreads*, while the read date is set by the user. LitRec also includes the global rating given to the book, a book summary, and book content. Book content was Part-Of-Speech tagged. Like Movielens, LitRec is also prepared for cross-validation and users are anonymous.

As can be observed LitRec has more items, but fewer ratings than Movielens. Consequently, sparsity is higher for LitRec than for Movielens. Sparsity is calculated as shown in Equation 2.

$$Sparsity = 1 - \frac{nonzeros\ entries}{total\ entries} \quad (2)$$

Figure 1 shows the distribution of the number of ratings per item and Figure 2 shows the distribution of the number of ratings by rating. As can be observed the number of ratings per item is closer to a straight line, contrary to what happens with Movielens. The number of ratings per rating follows the same distribution and 4 is the rating given to more items and 1 is the rating given to fewer number of items in both cases.

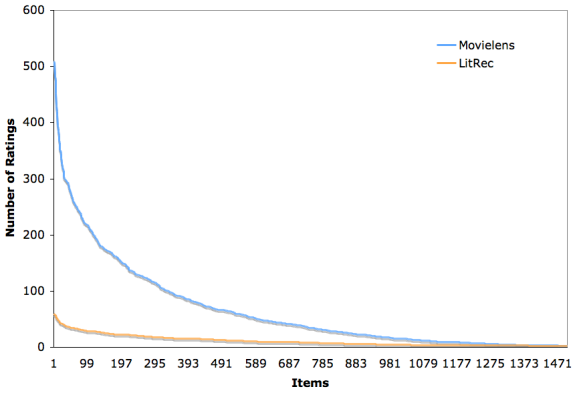

Figure 1: Number of ratings per item.

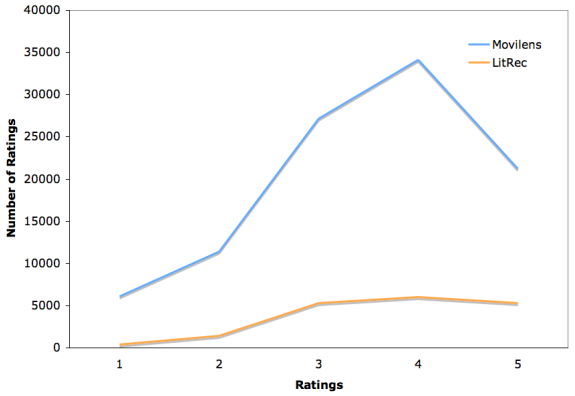

Figure 2: Number of ratings per rating.

4.2 Evaluation Metric

To evaluate our results we used the mean absolute error (MAE). MAE measures how far predictions are from the observed values. The MAE is described in Equation 3(where p_i is the predicted rating, o_i is the observed rating for item i and N is the number of rating-prediction pairs).

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|p_i - o_i| \quad (3)$$

4.3 Experimental Results

We tested our CF algorithm in both data sets. We considered between 1 and 100 item neighbors, using the different item representations. The neighborhood size in a kNN approach is a key factor for the algorithm performance. Figures 3, 4, and 5 show the results. As was expected, Movielens behavior is much smoother than LitRec's. This is explained by the distribution of ratings between users and items. We can also observe that, contrary to Movielens, LitRec achieves better values for MAE with smaller neighborhoods.

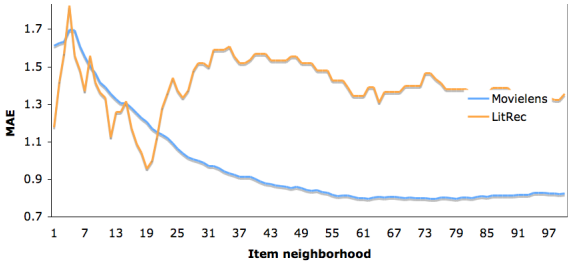

Figure 3: MAE for user vector item representation.

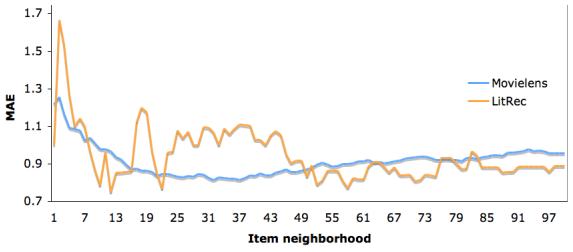

Figure 4: MAE for LDA item representation.

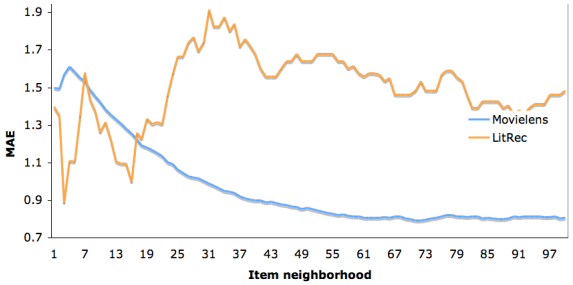

Figure 5: MAE for LSA item representation.

4.4 Item Variation

In these experiments we varied the number of items in both data sets and used user vectors for item representation. Firstly, we selected the 1682 books from LitRec and removed the others, in order to compare it with Movielens.

LitRec behavior remains consistent with the previous experiments (Figure 3), although the MAE slightly degraded, as shown in Figure 6.

Secondly, we observed that the LitRec book with more ratings only contained 60 ratings, so we removed all the movies from Movielens with more than 60 ratings and ran the CF algorithm for all the neighborhoods. The results are depicted in Figure 7. As expected, performance in both sets degraded although Movielens remained smoother than LitRec's performance. This may be explained by the different distribution of ratings between the two data sets.

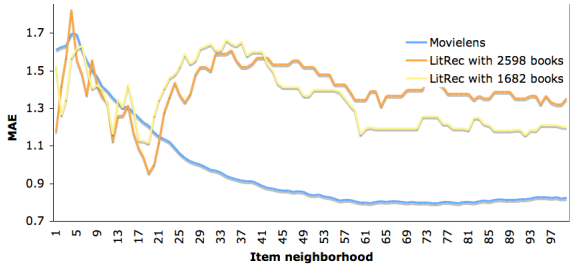

Figure 6: MAE using LitRec with the 1682 books.

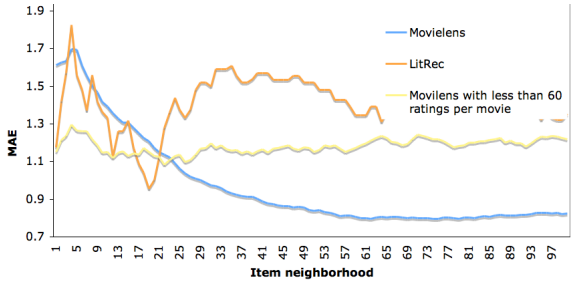

Figure 7: MAE using Movielens including only movies with fewer ratings.

5 CONCLUSIONS

Book recommendation is different from movie recommendation. Data sets are sparser making the CF task harder. This may be due to the fact that books are published in different languages while movies only have the original. For example, a Portuguese person who wants to rate the book "The Lord of the Rings" will most likely be rating the Portuguese translations, but if this person wants to rate the movie, he/she will be rating only the original version of the movie.

In this paper we present a comparative study between Movielens and LitRec. Movielens has been used in numerous studies and is considered by the research community to be a well formed data set. Nevertheless, book recommendation has specific recommendation problems that are not present in Movielens. Despite Movielens' qualities as a data set, it does not fit in all recommendation studies.

As was observed in the described experiments, although a CF approach has acceptable performance when using Movielens, even when the number of ratings per item is reduced, the same does not happen with LitRec due to rating distribution by user and by item. This suggests that other approaches should be tried to make book recommendation, e.g., using hybrid set-ups (CF + content-based filtering) or using only content-based filtering. LitRec has the advantage of containing several features (book author, genre, category, read, and rating date) and book content. Item content is always hard do get due to copyright restrictions.

Globally, LitRec performance is worse then Movielens performance in a CF set-up, even when items with the highest number of ratings were withdrawn from the data set. This confirms the conclusions achieved in previous works with LitRec that other book features are important to improve book recommendations accuracy (Vaz et al., 2012). Recommendation results, using LitRec, can take advantage from hybrid recommendation set-ups.

Using *Project Gutenberg* documents can pose a problem, because books are not recent. Despite the fact that books like "Romeo and Juliet" and "Sense and Sensibility" are always read, because they are classics, results can be biased towards users with given type of preferences. To generalize conclusions further analysis of results must be conducted. Nevertheless, LitRec can be used to study the literary book recommendation problem.

ACKNOWLEDGEMENTS

This work was supported by national funds through FCT - Fundação para a Ciência e a Tecnologia, under project PEst-OE/EEI/LA0021/2011.

REFERENCES

Bellegarda, J. R. and Juang, B. H. (2006). *Latent Semantic Mapping: Principles And Applications (Synthesis Lectures on Speech and Audio Processing)*. Morgan & Claypool Publishers.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.

Celma, O. (2010). *Music Recommendation and Discovery: The Long Tail, Long Fail, and Long Play in the Digital Music Space*. Springer Publishing Company, Incorporated, 1st edition.

Goldberg, K., Roeder, T., Gupta, D., and Perkins, C. (2001). Eigentaste: A constant time collaborative filtering algorithm. *Inf. Retr.*, 4(2):133–151.

Vaz, P. C., Martins de Matos, D., Martins, B., and Calado, P. (2012). Improving a hybrid literary book recommendation system through author ranking. In *Proceedings of the 12th ACM/IEEE-CS joint conference on Digital Libraries*, JCDL '12, pages 387–388, New York, NY, USA. ACM.

Ziegler, C.-N., McNee, S. M., Konstan, J. A., and Lausen, G. (2005). Improving recommendation lists through topic diversification. In *Proceedings of the 14th international conference on World Wide Web*, WWW '05, pages 22–32, New York, NY, USA. ACM.