Understanding the Temporal Dynamics of Recommendations across different Rating Scales

Paula Cristina Vaz[1], Ricardo Ribeiro[1,2], and David Martins de Matos[1,3]

[1]INESC-ID, Rua Alves Redol, 9, 1000-029 Lisboa, Portugal
[2]Instituto Universitário de Lisboa (ISCTE-IUL), 1649-026 Lisboa, Portugal
[3]Instituto Superior Técnico (IST), 1049-001 Lisboa, Portugal
{paula.vaz,ricardo.ribeiro,david.matos}@inesc-id.pt

Abstract. Libraries have large growing book collections and users have difficulty in browsing the whole collection when choosing new books to read, particularly when looking for books without a defined goal. In this case, recommendation systems are useful and play an important role in improving library usability. Recommendations are based on ratings and the quality of recommendations depends on the quality of the ratings. Studies show that users rate more items if scales have smaller granularity. In this paper, we propose a different rating scale for the book recommendation scenario in a collaborative filtering set-up and study how time influences rating relevance. Our findings suggest that the collaborative filtering algorithm benefits from a rating scale with smaller granularity. Moreover, if some conditions are met, rating prediction quality can be improved if we give lower weight to older ratings.

Keywords: Book recommendation, Collaborative Filtering, Temporal relevance, Rating scale

1 Introduction

Libraries both physical and digital have large growing book collections. Library users have difficulty in browsing the whole collection when choosing new books to read, particularly when looking for books without a defined goal. In this case, recommendation systems come in hand and play an important role in improving library usability.

Recommendation systems (RS) try to *know* the users observing their rating history. RS learn how the users rate their books and searches for other users with the similar tastes to generate reading recommendations. Two main techniques are used to develop recommendation systems [1]: content-based (CB) techniques in which users will be recommended items similar to those the user liked in the past; and collaborative filtering (CF) in which users will be recommended items that were preferred together. Each technique has limitations when taken individually, such as limited content analysis, the new item problem, sparsity, among others. To address these limitations, hybrid recommender systems have

been proposed where CB and CF techniques are combined in order to overcome the limitations of each technique.

To make suggestions, RS heavily depend on ratings, because only ratings tell the system what was the user opinion about an item. Ratings can be obtained implicitly or explicitly. Implicit ratings do not need any kind of user feedback. On the other hand, explicit ratings require explicit user feedback. Typically, these ratings are expressed on a 1-5- or 1-10-scale. This rating system has the advantage of better express users feelings about the book, but has the disadvantage of depending on user's explicit feedback. To further complicate maters, user preferences and opinions change over time. A user will most probably change the rating given to a book if that user is asked to rate it again.

This paper aims to (a) compare 1-5-scale rating to a like/neutral/dislike-scale in the rating prediction task; and (b) study the influence of rating age in predictions in the book recommendation scenario.

This paper is structured as follows: Section 2 gives an overview of the collaborative filtering approach. In section 3 we describe the data-set on which we based our experiments and the evaluation protocol. Section 4 describes and discusses the experiments. Section 5 describes related work. Finally, section 6 draws the conclusions and points to future directions.

2 Collaborative filtering

Following the work of [6], where the author proposes an user-based evolutionary kNN CF algorithm, in which ratings are weighted according to their age, we adapted the item-based CF algorithm in [7] by incorporating temporal information. Our temporal item-based CF algorithm (TICF) implements temporal decay through the use of the function in equation 1.

$$f_{u,i}^{\alpha}(t) = e^{-\alpha(t-t_{u,i})} \tag{1}$$

where u and i are the user and item relative to the rating, t is a time-stamp, and α controls the decaying rate. When α is set to 0, the time influence is ignored. $f_{u,i}^{\alpha}(t)$ measures the relevance of each observed rating $r_{u,i}$ in recommendation making, at time t based on the parameter α.

Temporal relevance has two dimensions: the age of the ratings given by the active user u, i.e., the user for whom recommendations are being made and the ratings given by the community, i.e., other users in the data-set. The age of the user ratings is controlled by parameter α. This parameter affects the rating prediction in equation 2, where $s_{i,j}$ is the adapted Pearson similarity (equation 3) between item i and item j and $r_{u,j}$ is the rating given by the active user u to item j.

$$P_{u,i} = \frac{\sum_j^k \; s_{i,j} * f_{u,j}^{\alpha}(t) * r_{u,j}}{\sum_j^k \; s_{i,j} * f_{u,j}^{\alpha}(t)} \tag{2}$$

The age of community ratings is controlled by parameter β that affects the item similarity calculation, as shown in equation 3, where $\overline{r^{\beta}}_i$ is the avearge

rating given by users to item i and each rating is affected by the time weight. If α and β are 0, the algorithm works as the usual item-based CF.

$$s(i,j) = \frac{\sum_{u=1}^{n}(f_{u,i}^{\beta}(t) * r_{u,i} - \overline{r^{\beta}}_i)(f_{u,j}^{\beta}(t) * r_{u,j} - \overline{r^{\beta}}_j)}{\sqrt{\sum_{u=1}^{n}(f_{u,i}^{\beta}(t) * r_{u,i} - \overline{r^{\beta}}_i)^2}\sqrt{\sum_{u=1}^{n}(f_{u,j}^{\beta}(t) * r_{u,j} - \overline{r^{\beta}}_j)^2}} \quad (3)$$

3 Evaluation protocol

For our experiments, we used the LitRec [9] data-set, from which we selected the 943 users with more than 10 ratings. This user selection left the data-set with 1,679 books and 34,156 ratings. LitRec was collected over a period of 5 years from *GoodReads.com* and was divided in a 90%-10% train-test-set. For each user in the test set, we selected the 10% most recent rated books. Then, we predicted a rating for each pair $<user, book>$ in the test-set and calculated the mean absolute error (MAE) as shown in equation 4, where p_i is the predicted rating, o_i is the observed rating for book i and N is the number of rating-prediction pairs.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|p_i - o_i| \quad (4)$$

4 Experimental set-up

We ran two experiments. First, we ran the TICF algorithm without using time relevance ($\alpha = \beta = 0$) to assess if the algorithm could benefit from a rating scale with smaller granularity. Then, we run the algorithm TICF with different values of α and β to study time influence in rating prediction quality.

4.1 Like/neutral/dislike rating scale

For this experiment, we converted the 1-5-scale of ratings in a 3-value-scale by replacing ratings 1-2 with a "dislike", rating 3 with a "neutral", and ratings 4-5 with a "like". This scale division was based on the reading of a significant number of reviews in the GoodReads.com site, that allowed us to get a sense of how users apply the rating scale in this particular data-set. We wanted to assess if error in rating predictions decreases with a smaller scale. The intuition behind the use of this type of scale is that it is easier for users to remember if they liked or hated a book, then to remember if they liked it with an intensity of 4 or 5. Moreover, according to the work presented by [8], users give more feedback to the system if the granularity of the rating scale is smaller. Then, we run the TICF algorithm on the data-set with $\alpha = \beta = 0$, varying the neighborhood size.

Figure 1 shows the error evolution. As can be observed, results are better for the 3-value scale. For the 1-5-scale, the MAE decreases until the 11^{th} neighbor and then becomes stabilized. For the 3-value-scale, the MAE decreases until the 4^{th} neighbor, then, increases until the 10^{th} neighbor, becoming steady after that.

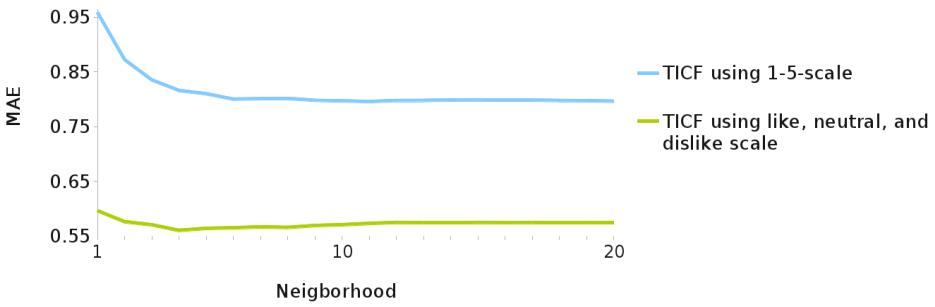

Fig. 1. MAE when using 1-5-scale ratings and a like, neutral, and dislike scale.

4.2 Temporal dynamics

In order to study the influence of rating age in prediction quality we run the TICF rating algorithm for different combinations of α and β with α and $\beta \in \{0, 0.1, ..., 1\}$. Rating age was measured in years and semesters and we ran the experiment for both rating scales. Figures 2 and 3 show the evolution of the MAE according to α and β variations. As can be observed, overall results are better when the rating age is considered in years (figure 2). Moreover, as expected, the MAE is lower for the 3-value rating scale (figures 2 (b) and 3 (b)).

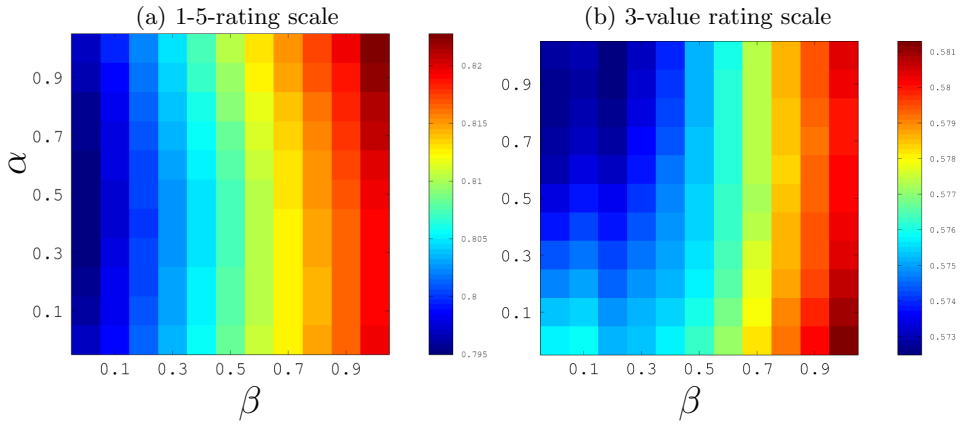

Fig. 2. MAE evolution for α and β variations, considering rating age in years.

Figures 2 and 3 show prediction quality changes with α and β variations. When the 1-5-scale is used, the MAE increases with the increment in the value of β, but when $\beta = 0$, the MAE decreases for $\alpha \in [0.3..0.8]$ (figure 2 (a)) and for $\alpha \in [0.1..0.3]$ (figure 3 (a)). These results show that rating prediction quality is affected both by recent and older ratings, regarding the community rating age. Regarding the active user rating age, rating prediction quality can be improved

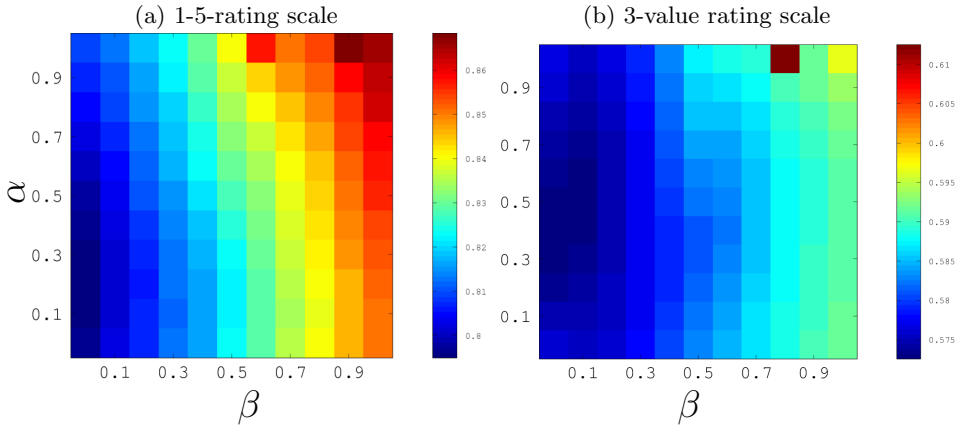

Fig. 3. MAE evolution for α and β variations, considering rating age in semesters.

if only the last two years of ratings are considered ($\alpha = 0.8$ for years and $\alpha = 0.6$ for semesters). The active user preferences are closer to his recent rated books than to older rated books.

When the 3-value rating scale is used, the MAE has the lowest values for $\alpha \in [0.8..1]$ and $\beta = 0.2$ (figure 2 (b)) and for $\alpha \in [0.3..0.6]$ and $\beta \in [0..0.1]$ (figure 3 (b)). results show that rating prediction quality can be improved if we consider ratings from the most recent four years ($\beta = 0.2$ for years and $\beta = 0.1$ for semesters). Regarding the active user rating age, rating prediction quality can be improved if only the present year of ratings is considered ($\alpha = 0.8$ for years and $\alpha = 0.6$ for semesters). The active user preferences are closer to the most recent rated books than to older ones. This scale is more sensitive to time relevance changes.

Results are consistent for both years and semesters and for both rating scales. Recall that the α parameter weights the active user ratings and that the β parameter weights the ratings of other users.

5 Related Work

Several approaches have been proposed to incorporate time relevance in the recommendation process. In [2] the authors adapt the item-based approach by incorporating time-based weights in the score prediction stage, but did not adapt similarity computation. [5] varies neighborhood size considering temporal information. [4] use matrix factorization to model changes in user and items over time. [6] adapted the a user-based CF algorithm by incorporating weights that give more relevance to recent ratings. Their approach affects the active user ratings and the community ratings. Nevertheless, the authors did not experiment with an item-based CF algorithm.

Rating scales used by recommendation systems have also been studied. In [3], the authors study the effect of different rating scales in user ratings, in a

cooking recipes recommendation system. The study shows that different users use rating scales differently and that wider scales are prone to more variability. In [8], the authors study the how often users rate across scales and conclude that as rating scale grows in granularity, users rate fewer items.

6 Conclusions & Future Work

Recalling the goals proposed at the beginning of the paper, we explored the TICF algorithm performance in predicting user tastes with a different rating scale. We converted the 1-5-scale in a like/neutral/dislike scale and run the TICF algorithm. Results shown that the TICF improves rating prediction quality when scale granularity decreases. From our study of the temporal relevance of rating age in the TICF algorithm, we were able to concluded that the active user preferences are closer to more recent ratings than older ones, especially considering a rating scale with lower granularity.

For future work, we want to confirm these results using other available datasets. We also want to explore if by giving less relevance to older rated books when using content-based recommendation, results confirm the ones obtained using a neighborhood-based CF algorithm.

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

References

1. G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):734–749, 2005.
2. Y. Ding and X. Li. Time weight collaborative filtering. In *Proceedings of the 14th ACM CIKM '05*, pages 485–492, New York, NY, USA, 2005. ACM.
3. C. Gena, R. Brogi, F. Cena, and F. Vernero. The impact of rating scales on user's rating behavior. In *Proceedings of the 19th UMAP'11*. Springer-Verlag, 2011.
4. Y. Koren and R. M. Bell. Advances in collaborative filtering. In *Recommender Systems Handbook*, pages 145–186. Springer, 2011.
5. N. Lathia, S. Hailes, and L. Capra. Temporal collaborative filtering with adaptive neighbourhoods. In *Proceedings of the 32nd ACM SIGIR '09*. ACM, 2009.
6. N. N. Liu, M. Zhao, E. Xiang, and Q. Yang. Online evolutionary collaborative filtering. In *Proceedings of the 4th ACM RecSys '10*, pages 95–102. ACM, 2010.
7. B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th WWW'01*. ACM, 2001.
8. E. I. Sparling and S. Sen. Rating: how difficult is it? In *Proceedings of the 5th ACM RecSys '11*, pages 149–156. ACM, 2011.
9. P. C. Vaz, D. Martins de Matos, B. Martins, and P. Calado. Improving a hybrid literary book recommendation system through author ranking. In *Proceedings of the 12th ACM/IEEE-CS JCDL '12*, pages 387–388. ACM, 2012.