On the use of Weighted Mean Absolute Error in Recommender Systems

S. Cleger-Tamayo
Dpto. de Informática.
Universidad de Holguín, Cuba
sergio@facinf.uho.edu.cu

J.M. Fernández-Luna & J.F. Huete
Dpto. de Ciencias de la Computación e I.A.
CITIC – UGR Universidad de Granada, Spain
{jmfluna,jhg}@decsai.ugr.es

ABSTRACT

The classical strategy to evaluate the performance of a Recommender System is to measure the error in rating predictions. But when focusing on a particular dimension in a recommending process it is reasonable to assume that every prediction should not be treated equally, its importance depends on the degree to which the predicted item matches the deemed dimension or feature. In this paper we shall explore the use of weighted Mean Average Error (wMAE) as an alternative to capture and measure their effects on the recommendations. In order to illustrate our approach two different dimensions are considered, one item-dependent and the other that depends on the user preferences.

1. INTRODUCTION

Several algorithms based on different ideas and concepts have been developed to compute recommendations and, as a consequence, several metrics can be used to measure the performance of the system. In the last years, increasing efforts have been devoted to the research of Recommender System (RS) evaluation. According to [2], "the decision on the proper evaluation metric is often critical, as each metric may favor a different algorithm". The selected metric depends on the particular recommendation tasks to be analyzed. Two main tasks might be considered: the first one, with the objective of measuring the capability of a RS to predict the rating that a user should give to an unobserved item, and the second one is related to the ability of an RS to rank a set of unobserved items, in such a way that those items more relevant to the user have to be placed in top positions of the ranking. Our interest in this paper is the measurement of the capability of a system to predict user interest in an unobserved item, so we focus on *rating prediction.*

For this purpose, two standard metrics [3, 2] have been traditionally considered: the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE). Both metrics try to measure which might be the expected error of the system, RMSE being more sensitive to the occasional large error: the squaring process gives higher weight to very large errors. A valuable property of both metrics is that they take their values in the same range as the error being estimated, so they can be easily understood by the users.

But, these metrics consider that the standard deviation of the error term is constant over all the predictions, i.e. each prediction provides equally precise information about the error variation. This assumption, however, does not hold, even approximately, in every recommending application. In this paper we will focus on the weighted Mean Absolute Error, wMAE, as an alternative to measure the impact of a given feature in the recommendations[1]. Two are the main purposes for using this metric: On the one hand, as an enhanced evaluation tool for better assessing the RS performance with respect to the goals of the application. For example, in the case of recommending books or movies it could be possible that the accuracy of the predictions varies when focusing on past or recent products. In this situation, it is not reasonable that every error were treated equally, so more stress should be put in recent items. On the other hand, it can be also useful as a diagnosis tool that, using a "magnifying lens", can help to identify those cases where an algorithm is having trouble with. For both purposes, different features shall be considered which might depend on the items, as for example, in the case of a movie-based RS the genre, the release date, price, etc. But also, the dimension might be user-dependent considering, for example, the location of the user, the users' rating distribution, etc.

This metric has been widely used for evaluation of model performance in several fields as meteorology or economic forecasting [8]. But, few have been discussed about its use in the recommending field; isolately several papers use small tweaks on error metrics in order to explore different aspects of RS [5, 7]. Next section presents the weighted mean absolute error, illustrating its performance considering two different features, user and item-dependent, respectively. Lastly we present the concluding remarks.

2. WEIGHTED MEAN ABSOLUTE ERROR

The objective of this paper is to study the use of a weighting factor in the average error. In order to illustrate its functionality we will consider simple examples obtained using four different collaborative-based RS (using Mahout implementations): i) Means, that predicts using the average rat-

[1] A similar reasoning can be used when considering squared error, which yields to the weighted Mean Root Squared Error, wRMSE.

Acknowledgements: This work was jointly supported by the Spanish Ministerio de Educación y Ciencia and Junta de Andalucía, under projects TIN2011-28538-C02-02 and Excellence Project TIC-04526, respectively, as well as the Spanish AECID fellowship program.

ings for each user; ii) LM [1], following a nearest neighbors approach; iii) SlopeOne [4], predicting based on the average difference between preferences and iv) SVD [6], based on a matrix factorization technique. The metric performance is showed using an empirical evaluation based on the classic MovieLens 100K data set.

A weighting factor would indicate the subjective importance we wish to place on each prediction, relating the error to any feature that might be relevant from both, the user or the seller point of view. For instance, considering the release date, we can assign weights in such a way that the higher the weight, the higher importance we are placing on more recent data. In this case we could observe that even when the MAE is under reasonable threshold, the performance of a system might be inadequate when analyzing this particular feature.

The *weighted Mean Absolute Error* can be computed as

$$wMAE = \frac{\sum_{i=1}^{U}\sum_{j=1}^{N_i} w_{i,j} \times abs(p_{i,j} - r_{i,j})}{\sum_{i=1}^{U}\sum_{j=1}^{N_i} w_{i,j}}, \qquad (1)$$

where U represents the number of users; N_i, the number of items predicted for the i^{th}-user; $r_{i,j}$, the rating given by the i^{th}-user to the item I_j; $p_{i,j}$, the rating predicted by the model and $w_{i,j}$ represents the weight associated to this prediction. Note that when all the individual differences are weighted equally $wMAE$ coincides with MAE.

In order to illustrate our approach, we shall consider two factors, assuming that $w_{i,j} \in [0, 1]$.

• **Item popularity**: we would like to investigate whether the error in the predictions depends on the number of users who rated the items. Two alternatives will be considered:

- i+ The weights will put more of a penalty on bad predictions when an item has been rated quite frequently (the items has a high number of ratings). We still penalize bad predictions when it has a small number of ratings, but we do not penalize as much as when we have more samples, since it may just be that the limited number of ratings do not provide much information about the latent factors which influence the users ratings. Particularly, for each item I_i we shall consider its weight as the probability that this item were rated in the training set, i.e. $w_i = pr(I_i)$.
- i– This is the inverse of the previous criterion, where we put more emphasis on the predictions over those items with fewer ratings. So the weights are $w_i = 1 - pr(I_i)$.

• **Rating distribution**: It is well known that the users does not rate the items uniformly, they tend to use high-valued ratings. By means of this feature we can measure whether the error depends on the ratings distribution or not. Particularly, we shall consider four different alternatives:

- rS+ Considering the overall rating distribution in the system, putting more emphasis on the error in the predictions on those common ratings. So the weights are $w_i = pr_S(r_i)$, r_i being the rating given by the user to the item I_i.
- rS– Inversely, we assess more weight to the less common ratings, i.e. $w_i = 1 - pr_S(r_i)$.
- rU+ Different users can use a different pattern of rating, so we consider the rating distribution of the user, in such a way that those common ratings for a particular user will have greater weights, i.e. $w_i = pr_U(r_i)$.
- rU– The last one assigns more weight to the less frequent rating, i.e. $w_i = 1 - pr_U(r_i)$.

Figures 1-A and 1-B present the absolute values of the MAE and wMAE error for the four RSs considered in this paper. Figure 1-A shows the results where the weights are positively correlated to the feature distribution, whereas Figure 1-B presents the results when they are negatively correlated. In this case, we can observe that by using wMAE we can determine that error is highly dependent on the users' pattern of ratings, and weaker when considering item popularity. Moreover, if we compare the two figures we can observe that all the models perform better when predicting the most common ratings. In this sense, they are able to learn the most frequent preferences and greater errors (bad performance) are obtained when focusing on less frequent rating values. Related to item popularity these differences are less conclusive. In some sense, the way in which the user rates an item does not depend of how popular the item is.

2.1 Relative Weights vs. Relative Error

Another different alternative to explore the benefits of using the wMAE metric is to consider the ratio between wMAE and MAE. In this sense, denoting as $e_{i,j} = abs(p_{i,j} - r_{i,j})$, we have that wMAE/MAE is equal to

$$wMAE/MAE = \frac{\sum_{i,j} w_{i,j}e_{i,j} / \sum_{i,j} e_{i,j}}{\sum_{i,j} w_{i,j}/N}.$$

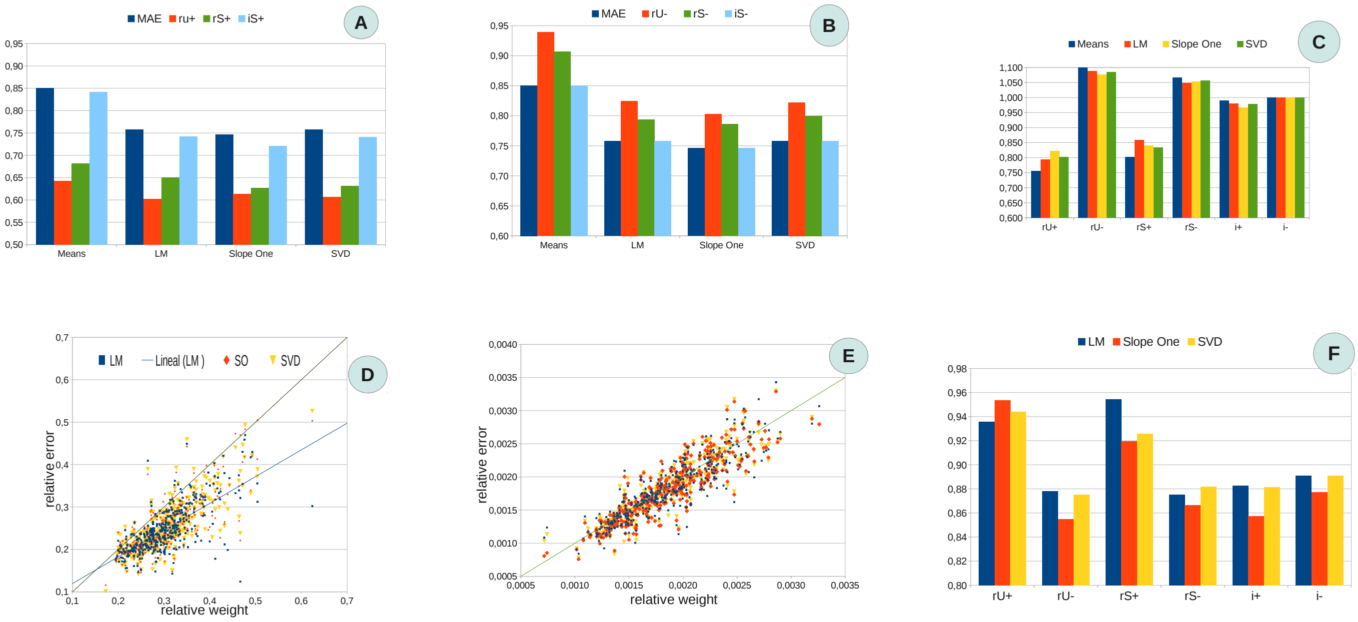
Taking into account that we restrict the weights to take its value in the $[0, 1]$ interval, the denominator might represent the average percentage of mass of the items that is related to the dimension under consideration whereas the numerator represents the average percentage of the error coming from this feature. So, when wMAE > MAE we have that the percentage of error coming from the feature is greater than its associated mass, so the the system is not able to predict properly such dimension. When both metrics are equal this implies that the expected error is independent of the feature.

In Figure 1-C we present the values of the wMAE/MAE where, again, we can see that there exists a dependence between the rating and the error. The error associated to the common ratings are less than the relative importance of this feature in the system whereas for less common ratings the system is not able to perform good predictions, being greater the relative error than its associated weights. This situation does not hold when considering item popularity.

Figures 1-D and 1-E present an scatter plot that relates the relative weights (horizontal axis) to the relative error (vertical axis) for each user in the system and for each RS used[2]. Particularly, in Figure 1-D we are considering rU+ as weighting factor. Note that, since there are 5 possible ratings, the relative weight is equal to 0.2 when all the ratings are equally probable and its value increases with the importance of the most used ratings. In this figure we can see that both percentage of mass and the percentage of error are positively correlated, being wMAE/MAE < 1 for most of the users. Moreover, there is a trend to improve the predictions for those users with higher relative mass (for example, we can see how the regression line for the LM model

[2]We have included all the users with at least 10 predictions.

Figure 1: Using wMAE in recommendations: absolute and relative values.



gets further away[3] from the line y=x). In some way we can conclude that recommendation usefulness of the rating distribution is consistent for all the users and RS models. On the other hand, Figure 1-E considers i+ as weights. In this case, although weights and error are positively correlated, there exists significant differences between different users. This result is hidden in the global measures.

2.2 Relative Comparison Among Models

Although wMAE might give some information about how the error has been obtained, there is no criterion about what a good prediction is. In order to tackle this situation we propose the use of the relative rather than the absolute error, i.e. the weighted Mean Absolute Percentage Error, wMAPE. Then, given two models, $M1$ and $M2$, the relative metric is defined as $wMAE_{M1}/wMAE_{M2}$. In this metric, the less the value, the greater the improvements. Thus, if we fix the model M2 to be a simple model (as the average rating) we obtain the wMAE values in Figure 1-F. From these values, we can obtain some conclusions as for instance that LM fits better the common user's preferences (rU+), whereas Slope One and SVD are more biased toward the overall rating distribution in the system (rS+). Similarly, we found that better improvements, with respect to the average ratings, are obtained when focusing on less frequent ratings. Finally, with respect to item popularity all the models obtain better improvements when considering the most popular items, although these differences are less significant.

3. CONCLUSIONS

In this paper we have explored the use of weighted Mean Average Error as a means to measure the RS's performance by focusing on a given dimension or feature, being able to uncover specific cases where a recommendation algorithm may be having suboptimal performance. This is a very useful way to know the origin of the errors found in the recommendations and therefore useful for improving the RSs, although its main problem is that it is not absolute as MAE.

[3]The other models perform similarly, but we have decided to not include these regression lines due to clarity reasons.

4. REFERENCES

[1] S. Cleger-Tamayo, J.M. Fernández-Luna and J.F. Huete. *A New Criteria for Selecting Neighborhood in Memory-Based Recommender Systems.* Proc. of 14th CAEPIA'11, pp. 423-432. 2011.

[2] A. Gunawardana and G. Shani. *A Survey of Accuracy Evaluation Metrics of Recommendation Tasks.* Journal of Machine Learning Research 10, pp. 2935-2962. 2009.

[3] J.L. Herlocker, J.A. Konstan, L.G. Terveen and J.T. Riedl. *Evaluating collaborative filtering recommender systems.* ACM Trans. Inf. Syst. 22, 1. 2004, pp. 5-53.

[4] D. Lemire and A. Maclachlan. Slope One Predictors for Online Rating-Based Collaborative Filtering. Proc. of SIAM Data Mining (SDM'05), 2005

[5] P. Massa and P. Avesani. Trust metrics in recommender systems. Computing with Social Trust, pp. 259-285 Springer 2009.

[6] B.M. Sarwar, G. Karypis, J. Konstan and J. Riedl. Incremental SVD-Based Algorithms for Highly Scaleable Recommender Systems. 5th International Conf. on Computer and Information Technology. 2002.

[7] T. Jambor and J. Wang. Goal-driven collaborative filtering: A directional error based approach. In Proc. ECIR'2010. pp. 407-419. 2010.

[8] C.J. Willmot. Statistics for the Evaluation and Comparison of Models. Journal of Geophysical Research, 90. pp. 8995-9005, 1985.