Empirical Assessment of a Collaborative Filtering Algorithm Based on OWA Operators

Miguel-Angel Sicilia,* Elena García-Barriocanal,† Salvador Sánchez-Alonso‡
Computer Science Department, University of Alcalá, Ctra. Barcelona km. 33.600, Alcalá de Henares, Madrid 28871, Spain

Classical collaborative filtering algorithms generate recommendations on the basis of ratings provided by users that express their subjective preference on concrete items. The correlation of ratings is used in such schemes as an implicit measure of common interest between users, that is used to predict ratings, so that these ratings determine recommendations. The common formulae used for the computation of predicted ratings use standard weighted averaging schemes as the fixed aggregation mechanism that determines the result of the prediction. Nonetheless, the surrounding context of these rating systems suggest that an approach considering a degree of group consensus in the aggregation process may better capture the essence of the "word–of–mouth" philosophy of such systems. This paper reports on the empirical evaluation of such an alternative approach in which OWA operators with different properties are tested against a dataset to search for the better empirical adjustment. The resulting algorithm can be considered as a generalization of the original Pearson formula based algorithm that allows for the fitting of the aggregation behavior to concrete databases of ratings. The results show that for the particular context studied, higher *orness* degrees reduce overall error measures, especially for high ratings, which are more relevant in recommendation settings. The adjustment procedure can be used as a general-purpose method for the empirical fit of the behavior of collaborative filtering systems.

1. INTRODUCTION

Recommender systems in e-commerce are aimed at helping customers by suggesting them products that could be of their interest, according to some algorithm that operates on navigation or purchase history or any other kind of data regarding products and customers. More specifically, *collaborative filtering* (CF) techniques[1,2,3] analyze preference data for the purpose of producing useful recommendations to customers. CF systems proceed by first matching the target user against the historical

*Author to whom all correspondence should be addressed: e-mail: msicilia@uah.es.
†e-mail: elena.garciab@uah.es.
‡e-mail: salvador.sanchez@uah.es.

INTERNATIONAL JOURNAL OF INTELLIGENT SYSTEMS, VOL. 23, 1251–1263 (2008)
 Published online in Wiley InterScience (www.interscience.wiley.com). • DOI 10.1002/int.20318

user database to discover *neighbors*—i.e., users that have historically had similar preferences—and then recommending products that neighbors like, since it is assumed that the target user will "probably" also like them.[4] Other recommendation approaches are content based, i.e., they use some kind of semantic representation of the product descriptions and use them as a source of similarities for the task of selecting recommendations. Content-based and preference-based techniques are complementary, as demonstrated in existing recommender systems, e.g., Refs. 5–7. However, the approach of pure rating-based schemes can be approached independently, since the assumptions of these methods are complementary to content-based ones.

The rationale behind collaborative filtering algorithms has been said to be the automation of the process of "word of mouth," by which people recommend products or services to others with similar taste,[2] so that preferences (either explicitly or implicitly collected) are the main source for recommendations. But in most current e-commerce systems, customers are not informed about the identity of their *neighbors*, so that "reputation" in trusting recommendations is not exploited, and in fact, it would be almost impossible to use in practice, due to the large population of users and the (relatively) generalized unwillingness to reveal oneself's identity. In consequence, the mathematical models used to predict user preferences only deal with past-recorded preferences, which are in most cases expressed in numerical scales, e.g., $\{1, \ldots, 5\}$ or $[1, 5]$. Such preferences reflect the subjective interest of users on concrete items.

While existing models of collaborative filtering have been successfully used in e-commerce, their mathematical models reported in the scientific literature are based on intuitions about the idea of "word-of-mouth" recommendation that deserve more thorough studies. For example, recent work has provided evidence that the consideration of bipolarity affects the results and user acceptability of item predictions.[8] Furthermore, assuming that reputation is not considered, it is reasonable to initiate empirical inquiry instead of adhering to mathematical models prescribed a priori. In other words, the algorithm that aggregates existing ratings to produce predictions can be extended with more flexible formulations. This is the main motivation of the present study.

In this paper, we focus on the aggregation of the ratings of the neighbors in classical collaborative filtering models, continuing the direction described in Ref. 9. The problem can be succinctly stated in the following way: The system has to decide to recommend or not an item i (e.g., a book or a movie) to a concrete user u_j, and the overall population U of users and their ratings to items are used to find users with similar preferences as u_j ("neighbors"). Then, the aggregation of ratings of neighbors represents the contribution of the ratings of the users in U to the prediction of the rating for the user u_j to item i. This study focuses on the effect of different degrees of fuzzy majority[10] as an alternative to classical models, even though other concrete designs could be hypothesized to be a better model than averaging. Concretely, the use of OWA operators as part of the computation of predictions is approached as a generalization of the classical Pearson model of CF. Instead of using an OWA operator a priori, the method described uses an empirical adjustment procedure that can be fully automated for real-world settings.

The rest of this paper is structured as follows. Section 2 provides the extended algorithm proposed and its rationale, and succinctly sketches the technology and design of the software used in the present study. Then, Section 3 describes the evaluation setting and the measures used. Section 4 reports and discusses the results of the study and the final design obtained for the concrete dataset. Finally, conclusions and future research directions are provided in Section 5.

2. GENERALIZING THE PEARSON ALGORITHM WITH OWA OPERATORS

The main idea of the study reported here is that of comparing the rating aggregation scheme used in classical collaborative filtering[11] with other alternatives that represent different aggregation behaviors.

The mathematical models used for the comparison are based on the classical *GroupLens* heuristic described in the seminal paper.[3] This original model consisted of two computation steps. First, a correlation measure in $[-1, 1]$ is computed as a measure of similarity of user preferences. Correlation coefficients (between each pair of users a and b) are in the form described in (1), being $v_{x,y}$ the explicit rating (i.e., a rating explicitly provided by the user) and $\overline{v_x}$ is the average rating of user x.

$$w(a,b) = \frac{\sum_{j\in\mathcal{I}}(v_{a,j} - \overline{v}_a)(v_{b,j} - \overline{v}_b)}{\sqrt{\sum_{j\in\mathcal{I}}(v_{a,j} - \overline{v}_a)^2 \sum_{j\in\mathcal{I}}(v_{b,j} - \overline{v}_b)^2}} \tag{1}$$

Then, in the second step, expression (2) shows the model for predicting the rating to item l by user u.

$$p_{u,l} = \overline{v_u} + \frac{\sum_{i\in\mathcal{U}}(v_{i,l} - \overline{v_i})w(u,i)}{\sum_{i\in\mathcal{U}}|w(u,i)|} \tag{2}$$

From this classical scheme, the idea of the inquiry reported in this paper is that of substituting the original weighted aggregation expression with a flexible operator. The OWA operator[12] has been selected for this study because it is acknowledged as a model for processes of consensus reaching in groups, which appears close to the problem at hand. OWA-based recommendation models have been proposed,[13] but no empirical study has been reported to date that compares their properties in practical settings with existing classical CF approaches.

Expression (3) shows the model to predict the rating to item l by user u, where the difference from the original schema is that the original averaging of influence in expression (2) has been changed to an aggregation operator $\mathcal{A}$.

$$p_{u,l} = \overline{v_u} + \mathcal{A}[(v_{1,l} - \overline{v_1})w(u,1), \ldots, (v_{i,l} - \overline{v_i})w(u,i)] \quad i \in U \tag{3}$$

In the case of using the OWA as the aggregation operator, we have the following expression:

$$p_{u,l} = \overline{v_u} + \mathcal{OWA}[(v_{1,l} - \widehat{\overline{v_1}})w(u,1), \ldots, (v_{i,l} - \widehat{\overline{v_i}})w(u,i)] - 0.5 \quad i \in U \tag{4}$$

The inputs to the OWA are normalized (denoted for each input x as $\widehat{x}$), so the operator yields a value in [0,1] representing the deviation. Each input value is thus in the form $\widehat{(v_{i,l} - \overline{v_i})w(u,i)}$. The subtraction of 0.5 is required to preserve the original properties of the model, given that OWA results are nonnegative. This is because the scale [0,1] should be changed to a bipolar scale, in which, for example, a value of 0.3 actually represents a negative contribution, which becomes -0.2. Without that adjustment, all the predicted values would result higher than the average ratings of the user considered. This is a possible model that produces values similar to the original Pearson formula; however, other approaches could be considered, e.g., providing more weight to negative correlations as pointed out in Ref. 8.

We have chosen not to change the correlation coefficient in (1) to avoid changing its robust interpretation of matching profiles, so that it is expression (2) which becomes modified.

The *MovieLens* database and the software included in the *CoFe* distribution[a] have been used to implement the evaluation process. We have selected the OWA operator family (5) [14], so that

$$\mathcal{A}(x_1, \ldots x_n) = \sum_{j=1}^{n} x_{(j)} \cdot w_j \tag{5}$$

where $x_{(j)}$ are an increasing permutation of the input variables, and the vector of weights $W = (w_1, \ldots w_n)^T$ satisfies $w_i \in [0,1]$ and $\sum_i w_i = 1$. Since the number of input values n depends on the form of computation of neighborhoods in the collaborative filtering algorithm, the evaluation procedure consisted of computing the adequacy of sets of OWA operators for a given *orness* ϕ, for each possible number of input values (the maximum for this can be configured in CoFE), so that the evaluation proceeds for sets of operators that for identification purposes will be denoted as $\mathcal{A}^{\phi}$. The orness is a chosen significant parameter for adaptation. It is significant because it allows to find the (possibly)best model in the spectrum of possibilities of aggregation offered by the OWA family of operators.

2.1. The Generalized Prediction Algorithm

The algorithm used is provided in high-level pseudocode in the following listing, in which $\mathcal{U}$ represents the user and $\mathcal{I}$ the item for which the prediction is being asked for. Ω is the orness required for the computation of the prediction.

PREDICT-RATING($\mathcal{U}$, $\mathcal{I}$, Ω)
1 *neighbors* $\leftarrow$ GETNEIGHBORGS($\mathcal{U}$)
2 $i \leftarrow 0$
3 **for** each $n_i \in$ *neighbors*
do

[a]http://eecs.oregonstate.edu/iis/CoFE/

$r \leftarrow$ GETRATING$(n_i, \mathcal{I})$
if ($r \neq$ *null*)
then
inputs[i] $\leftarrow$ GETSIMILARITY$(\mathcal{U}, n_i) \cdot (r -$ GETMEANRATING$(n_i))$
i++

$w \leftarrow$ GENERATEWEIGHTSFORORNESS(SIZE(*inputs*), Ω)
$\mathcal{A} \leftarrow$ CREATEOWAOPERATOR(w)
return $\mathcal{A}$(inputs) + GETMEANRATING$(\mathcal{U})$

The algorithm uses an intermediate data structure holding the inputs to the OWA, and then the operator is dynamically created depending on the number of inputs that would actually require integration. The sorting of the weights used to create the OWA operator thus introduces a $O(n \cdot ln)$ complexity step to the algorithm, but this is asymptotically nonrelevant since the GENERATEWEIGHTSFORORNESS requires the use of an equation solver. For fixed-length approaches to recommendation, this should be substituted by an approach in which that procedure is called only once at start-up, but this would require constraining the number of inputs used to a minimum threshold. This in turn entails that predictions below such minimum number of inputs could not be computed. This makes sense for databases as *MovieLens* that are fairly homogeneous, but not for a general case. An alternative would be that of the creation at initialization of a number of OWAs of different input sizes to deal with different cases.

The GENERATEWEIGHTSFORORNESS procedure for the particular implementation is based in the expressions of Liu and Chen.[15] Concretely, the operators produced are geometric OWAs (GOWA), which have a fixed ratio q for adjacent weights.

For a given $orness(W) = \Omega$ and n, q is the root of solution of (5).[b]

$$(n-1)\Omega q^{n-1} + \sum_{i=2}^{n}((n+1)\Omega - i + 1)q^{n-1} = 0 \tag{6}$$

Then, the GOWA weights can be expressed by (6).

$$w_i = \frac{q^{i-1}}{\sum_{j=0}^{n-1} q^j} \tag{7}$$

2.2. Implementation Issues

The empirical assessment described in the next section was realized on the *Collaborative Filtering Engine* (CoFE) software, version 0.3. CoFE runs as a server to generate recommendations for individual items, top-N recommendations over all items, or top-N recommendations limited to one item type. Recommendations are computed using the well-tested nearest-neighbor algorithm (Pearson's algorithm). User data are stored in a MySQL database.

[b]In the original paper, the summation is preceded by a minus sign, due probably to an erratum.

The Java libraries provided in source code version in CoFE provide an interface `CFAlgorithm` encapsulating the essential functionality of a collaborative filtering algorithm. The `CFAlgorithm` method of interest in our present research has the following signature:

```
public ItemPrediction predictRating  (int userID,
 int itemID)
 throws CFNotImplementedException;
```

Concretely, the overall pseudocode algorithm described above has been integrated in a variant version of the CoFE class `SimplePearsonAlgorithm`. The testing of different OWA operators has been realized through a private method with the following signature:

```
void generateWeightsForOrness (double[] weights,
 double orness);
```

The method generates the weights for the given `orness` value via the above-mentioned Liu and Chen computation procedure.[15] The *OR-objects* libraries[c] were used for the solving of the equation that computes parameter q. Concretely, the `drasys.or.nonlinear.EquationSolution` class was used with a straightforward Java implementation of the function encapsulated in a class `GOWAFunction` implementing the `drasys.or.nonlinear.FunctionI` interface. Such equation-solving class internally implements a bisection algorithm for solving single equations.

3. EVALUATION SETTING

The evaluation setting is based on a systematic assessment of overall properties of adjustment to the available empirical dataset. The dataset used is the original and well-known *MovieLens* database with 100,000 ratings (in the 1–5 range) from 943 users on 1682 movies. The data was collected through the MovieLens Web site (`movielens.umn.edu`) during the 7-month period from September 19, 1997 through April 22, 1998. These data were cleaned up—users who had less than 20 ratings or did not have complete demographic information were removed from this dataset.

The assessment of the appropriateness of OWA operators was carried out by using a variant of the well-known *All but 1* protocol,[16] in which a single rating of the database is selected, and the rest of the database is used to predict it. This emphasizes the evaluation of the algorithms after having a steady usage leading to a fairly significant database. This is coherent with the kind of adjustment process proposed here, because it would be done in practical situations after a database of reasonable size and coverage has been collected.

[c]http://opsresearch.com/

The approach for the evaluation was that of generating a OWA weighting vector of maximum entropy for the given n,[15] and then proceed to the systematic testing of different orness levels. The process of evaluation consisted in the following. For each explicit rating in the database, the following is done:

- The explicit rate is removed (or ignored).
- Then, the rest of the database is used for the prediction of that item, using the classical and modified schemes as described above.
- For each scheme, the prediction is compared to the original explicit rating, and the error is used to compute measures of overall predictive quality (described later) *for each scheme*.

This allows for a standard evaluation for the given evidence, and no cross-validation is required since the standard Pearson model does not require a process of adjustment to empirical data.

As measures of predictive quality, the MMRE and PRED(x) measures commonly used in software engineering were used. The rationale for this is that they provide two viewpoints on quality of prediction that are complementary. MMRE provides a global prediction quality measure, whereas PRED(x) provides a measure of the percentage of "high-quality predictions" (related to the x value). The definitions of these measures are provided in what follows.

Mean magnitude of relative error (MMRE) is defined as[17]

$$MMRE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{e_i - \widehat{e_i}}{e_i}\right| \tag{8}$$

where e_i is the actual value of the variable and $\widehat{e_i}$ its corresponding estimate, and n is the number of observations. Thus if MMRE is small, then the predictions can be considered as good.

Prediction at level p, where p is a percentage, is defined as the quotient of number of cases in which the estimates are within the p absolute limit of the actual values, divided by the total number of cases. For example, PRED(0.2) = 70 means that 70% of the cases have estimates within the 20% range of their actual values. Real numbers in the [0,1] interval will be used here to express PRED values.

The MMRE and PRED measures can be used for the assessment of overall adjustment properties. However, an additional consideration should be brought into the analysis. Since collaborative filtering algorithms are ultimately used for recommendation, the specifics of such process must be considered. It is commonly acknowledged that the most important errors to avoid in e-commerce recommendations are *false positives*—as pointed out in Ref. 4—since they may lead to "angry customers." In consequence, the evaluation has been carried out considering "layers" of prediction values, so that the MMRE and PRED measures have been collected *for each* threshold τ. In consequence, for a given τ value only the ratings that are above such value are considered in computing the measures. This allows us to focus, for example, "on item-rating predictions above 4" (supposed a scale from 1 to 5), which would be the predictions that would more likely result in recommendation to final users.

4. RESULTS AND DISCUSSION

The systematic gathering of measures is reported here by first considering a base case of orness with value 0.5 and then contrasting other orness values for different values of τ.

4.1. The Base Case Orness = 0.5

Figure 1 depicts the PRED(p) values for $p \in \{5, 10, 20, 30\}$ at different threshold values τ. The values of τ cover the rating domain at steps of 0.1, thus providing enough detail for our analysis. A plane at PRED value 0.5 has been provided for the ease of understanding of the figure.

Several important aspects of Figure 1 are worth noting for the subsequent analysis. First, PRED(30%) indicates that the algorithm behaves with good predictive capabilities with an approximate error of 1.5 points in the scale. This goodness of adjustment is confirmed by the MMRE value of 0.3 for the whole dataset. Another important aspect to be considered is that for "average" values $2 \lesssim \tau \lesssim 3$, the predictive capabilities are the best.

Figure 2 provides a view of the amount of overall error for the different values of τ. It also provides a percentage (expressed in the [0,1] interval) of the data used for the computation of the measures: (*size*), and overall positive and negative errors (i.e., predictions higher and lower than the actual ratings, respectively). An interesting fact that follows from the observation of that data is that negative error increases for items with higher ratings and positive error does the reverse. This could be considered as a "conservative" behavior, since it is biased to producing "low" values for medium-high ratings.

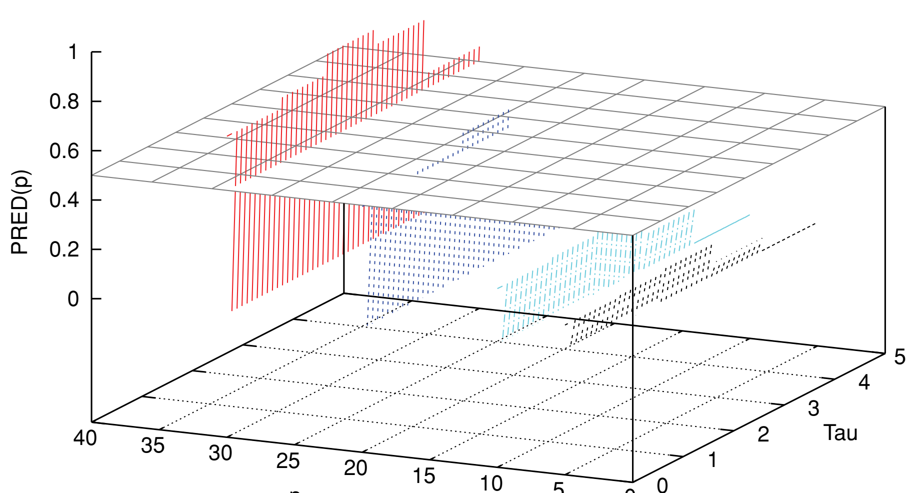

Figure 1. PRED(p) values for orness 0.5 at different τ values for $p \in \{5, 10, 20, 30\}$.

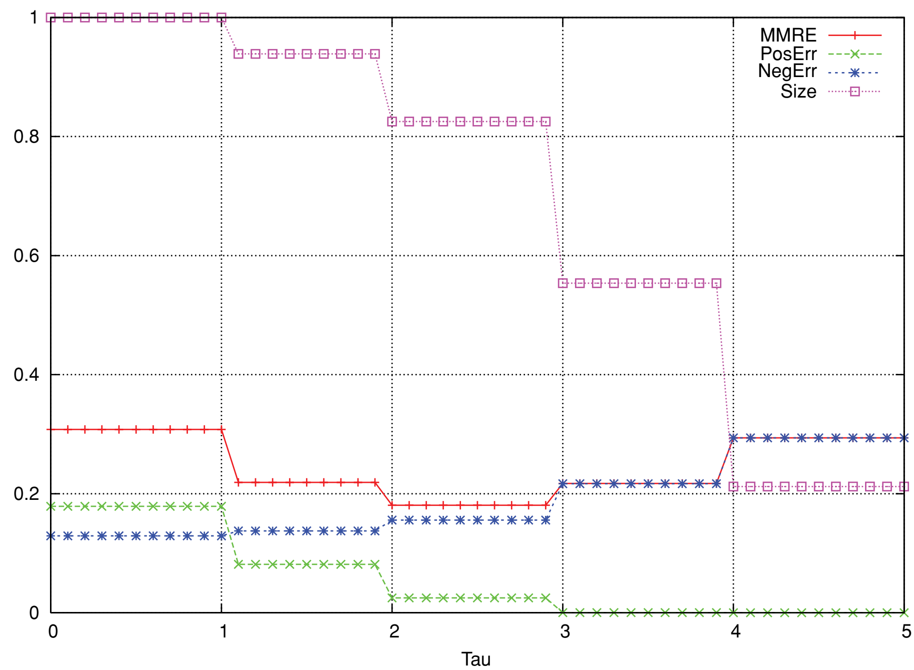

Figure 2. MMRE, positive error, negative error, and number of ratings used in calculating the measures for orness 0.5 at different τ values.

4.2. Comparison with the Base Case

The base case serves as a point of comparison for a process of search of better orness parameters. Since in principle there is not a clear guiding heuristic for the search, the procedure implemented for this study has followed a *divide-and-conquer* strategy, computing the adjustment measures first in one of the directions and, if unsuccessful, trying then with the other. In consequence, the algorithm started with orness 0.25. The results for the PRED measure are provided in Figure 3.

The overall distribution of predictive measures is similar to the base case, but the values are systematically worse, for all the cases with decreases below 10%.

MMRE measures are only slightly worse for the whole dataset (0.31), and the same effect occurring with positive and negative error remains.

This analysis points out that the search strategy should instead try with orness values above 0.75. The first value assessed is the intermediate 0.75. PRED(x) values are systematically better than the ones of the base case in an amount of approximately a 5%–10%, depending on the τ value. This points to a possible improvement in the direction of increasing orness. Data from several orness values are shown in Table I, on average for different τ values.

Table I provides a clear picture that for the dataset and with an analysis by τ levels, higher OWA values increase the goodness of fit. However, this overall picture

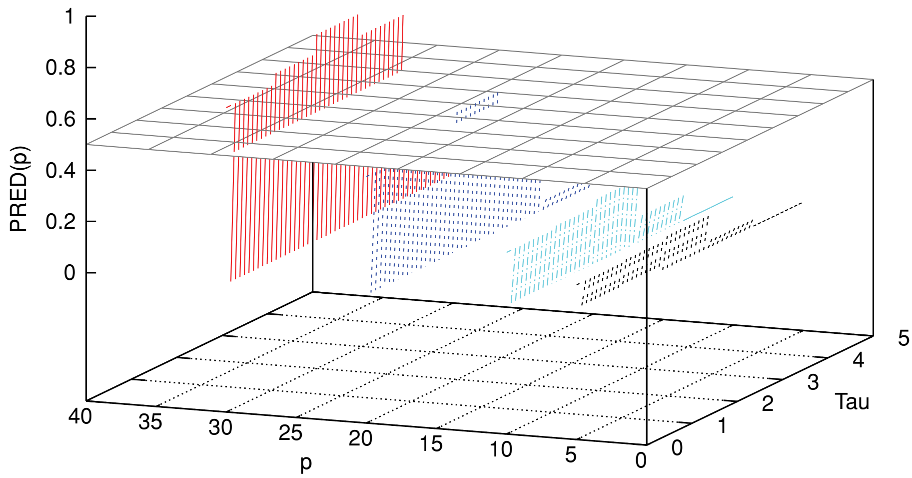

Figure 3. PRED(p) values for orness 0.25 at different τ values for $p \in \{5, 10, 20, 30\}$.

Table I. Average MMRE, PRED and error values for sample ornessvalues

Orness	$\overline{MMRE}$	$\overline{PRED}(30)$	$\overline{PRED}(20)$	$\overline{PRED}(10)$	$\overline{PosErr}$	$\overline{NegErr}$
0.5	0.25	0.72	0.42	0.17	0.06	0.19
0.75	0.23	0.78	0.48	0.2	0.06	0.17
0.99	0.21	0.83	0.56	0.25	0.07	0.15

must be balanced with an analysis of positive error (which entails the prediction of higher values, i.e., overratings) and its potential effects in the final recommendation of items.

The difference in average positive error between the p = 0.99 case and the base case amounts to 0.018, while concentrating on $\tau = 2$ it amounts 0.013. Consequently, the distribution of error does not concentrate in general terms on "medium-high ratings." An additional empirical analysis of the standard deviation of errors for that values confirms that uniform distribution of error. This can be interpreted in the sense that using higher OWA values improves the overall adjustment, probably introducing a reasonable balance of the original differences in positive and negative errors. This is complemented with the fact that the additional positive error does not concentrate on "medium-high values" so that the overall recommendation capability of the approach is increased. These results point out that or–like aggregation for this concrete dataset provides a better model for rating prediction.

5. CONCLUSIONS AND FUTURE WORK

Classic collaborative filtering algorithms can be generalized with more flexible models of aggregation. Such generalizations can be considered as parameterizations of the original models, which are thus subject to empirical adjustment of the parameters. The classical CF formulae are a possible approach that has demonstrated useful, but the changes proposed in this paper extend them by allowing a further level of adaptation: adapting the aggregation behavior. Thus, the resulting model is more flexible since it has an additional parameter for fitting the algorithms.

This paper has reported the reformulation of the classical Pearson algorithm in terms of an OWA operator, being the orness of the OWA the parameter considered. An implementation of the modified algorithm has been reported, and an empirical assessment using an *All but 1* approach has been reported.

The analysis showed that high orness values provide better overall predictive results and more reasonable positive and negative error balances.

The procedure used in this study can be implemented as an algorithmic process that could obtain an heuristic of the best OWA orness value for a given empirical database. Even though the process is complex in computing time (and thus not practical for continuous adjustment), it can be used as a periodic offline process for recommender systems. This fits with the nature of CF systems in which the overall recommendation capabilities are only expected to change with a significant increase of the database of ratings.

The alternative aggregation scheme provided in this paper complements the use of bipolarity described elsewhere,[8] but does not exhaust the range of possibilities, provided that there is a significant amount of variants and families of aggregation operators with diverse properties; see, for example, Ref. 18 This opens up a direction for research on testing diverse aggregation schemes to the empirical data. Consequently, further work should deal with other aggregation schemes. In addition, the adjustment problem addressed in this paper must be complemented with experimental studies on the suitability of the adjusted operators for the perceptions of users, if sound models or generalized hypotheses of the "word-of-mouth" paradigm are sought. In Ref. 8, an example of such an experimental procedure is provided.

Acknowledgments

We acknowledge the guidance of Prof. Enrique Herrera-Viedma in finding approaches for the generation of OWA operators for given orness values.

References

1. Resnick P, Iacovou N, Suchak M, Bergstrom P, Riedl J. GroupLens: An open architecture for collaborative filtering of netnews. In: Proc of ACM 1994 Conference on Computer Supported Cooperative Work, 1994. pp 175–186.
2. Shardanand U, Maes P. Social information filtering: Algorithms for automating "word of mouth." In: Proc of CHI'95—Human Factors in Computing Systems; 1995. pp 210–217.
3. Konstan J, Miller B, Maltz D, Herlocker J, Gordon L, Riedl J. GroupLens: Applying collaborative filtering to usenet news. Commun ACM 1997;40(3):77–87.
4. Sarwar BM, Karypis G, Konstan JA, Riedl J. Analysis of recommender algorithms for e-Commerce. In: Proc ACM e–Commerce Conf; 2000. pp 158–167.

5. Sarwar, B, Karypis G, et al. Item based collaborative filtering recommendation algorithms. In 10th International World Wide Web Conf Hong Kong; 2001.
6. Paulson P, Tzanavari A. Combining collaborative and content–based filtering using conceptual graphs. In: Lawry J, Shanahan JG, Ralescu A, editors. Modeling with words: Learning, fusion, and reasoning within a formal linguistic representation framework, LNAI Vol 2873. Berlin: Springer-Verlag; 2003. pp 168–185.
7. Bezerra BL, Carvalho FA. A symbolic approach for content-based information filtering. Inform Proc Lett; 2004;92:45–52.
8. Sicilia MA, García-Barriocanal E. On the use of bipolar scales in preference-based recommender systems. In: Proc of EC-Web; 2004. pp 268–276.
9. Sicilia MA, García-Barriocanal E. Empirical evaluation of alternatives to the aggregation of neighbor': ratings in social filtering. In: Proc of the International Summer School on Aggregation Operators; 2005. pp 123–127.
10. Pasi G, Yager RR. Modeling the concept of fuzzy majority opinion. In: Bilgic T, De Baets B, Kaynak O, editors. Fuzzy sets and systems–IFSA, Proc of the 10th International IFSA World Congress; 2003. pp 143–150.
11. Herlocker JL, Konstan JA, et al. An algorithmic framework for performing collaborative filtering. In: Proc of the 22nd Annual International ACM SIGIR Conference on Research and Development in information retrieval; 1999; Berkeley, CA.
12. Yager RR, Filev DP. Parameterized and-like and or-like OWA operators. Int J Gen Syst 1993;22:297–316.
13. Yager RR. Fuzzy logic methods in recommender systems. Fuzzy Sets Syst; 2003;136(2):133–149.
14. Yager RR. On ordered weighted averaging aggregation operators in multicriteria decision making. IEEE Trans Syst Man Cybern 1988;18:183–190.
15. Liu X, Chen L. On the properties of parametric geometric OWA operator. Int J Approx Reason 2004; 35:163–178.
16. Breese JS, Heckerman D, Kadie C. Empirical analysis of predictive algorithms for collaborative filtering. In: Proc of the 14th Annual Conf on Uncertainty in Artificial Intelligence; 1998. pp 43–52.
17. Conte SD, Dunsmore HE, Shen VY. Software engineering metrics and models. Menlo Park, CA: Benjamin/Cummings; 1986.
18. Calvo T, Kolesárová A, Komorniková M, Mesiar R. Aggregation operators: Basic concepts, issues and properties. In: Calvo T, Mayor G, Mesiar R, editors. Aggregation operators: New trends and applications. Studies in fuzziness and soft computing, Vol 97. Berlin: Springer; 2002. pp 3–106.

APPENDIX

To do the comparison of actual and predicted values, some indicators could be used. If V_e is the estimated value and V_a the actual value, the relative error (*RE*) and the error relative (*ER*) to the estimates are

$$RE = \frac{V_a - V_e}{V_a}$$

$$ER = \frac{V_a - V_e}{V_e}$$

Frequently is need to know the relative error for a set of estimators, for example, usually is desired to know whether the effort estimations done are accurate for a set of developed projects. The medium relative error (similar for the medium error relative) for a set of projects is

$$\overline{RE} = \frac{1}{n}\sum_{i=1}^{n} RE_i$$

Also is possible to calculate the value of these same indicators considering the absolute value $MMRE = \overline{|RE_i|}$, in this case for n projects the expression is

$$MMRE = \frac{1}{n}\sum_{i=1}^{n} |RE_i|$$

These concepts are used to define a measure for the prediction quality. For a set of n projects, i is the number of them whose medium relative error value is less or equal q

$$PRED(q) = \frac{i}{n}$$

The prediction of level q, $PRED(q)$ gives an indication of the adjustment degree for a data set, based on the value of the RE obtained for each date. For example, if $PRED(0.3) = 0.4$ the 40% of the projects have a medium relative error below 30%. To evaluate the performance of a given model could be considered that a good model is one with $MMRE \leq 0.25$ and $PRED(0.25) \geq 0.75$.