MISIR: RECOMMENDATION SYSTEMS IN A KNOWLEDGE MANAGEMENT SCENARIO

Carlos Eduardo Barbosa[1]
Jonice Oliveira[2]
Luciano Maia[1]
Jano Moreira de Souza[1,2]
1-IM/UFRJ - Computer Science Department - Institute of Mathematics / Federal University of Rio de Janeiro (UFRJ)
2-COPPE/UFRJ - Graduate School and Research in Engineering / Federal University of Rio de Janeiro (UFRJ)
PO Box 68.511 – ZIP 21.941-972 - Rio de Janeiro - Brazil
edu@ufrj.br, jonice@cos.ufrj.br, lucianomaia@gmail.com, jano@cos.ufrj.br

ABSTRACT

In a scientific scenario we can notice the predominance of explicit knowledge being manipulated and distributed, which makes recommender systems very useful in this environment. But along with a knowledge management approach this kind of system can support the organization in better identifying competences, help engage users in a continuous and dynamic knowledge exchange, and customize knowledge dissemination as much as possible. MISIR is a collaborative recommender system, developed to help to the process of knowledge exchange. In this work we detail a collaborative recommender system which is used in a Scientific Knowledge Management Environment; we show how this approach can be aimed at a KM process and how this approach can deal with other kinds of knowledge used in research centers and universities.

KEYWORDS

Recommendation Systems, Knowledge Management, Web Community.

1. INTRODUCTION

Observing the considerable and increasing amount of digital information available it becomes more and more difficult to discern which the most relevant information is. Unfortunately we do not have enough time to read, interpret and assimilate all data about a subject of our interest. We can try to optimize the work with the use of Recommendation Systems.

A Recommendation System is the evolution of Information Filtering. However, while the focus of Information Filtering approach is on the removal of non-relevant results from a search, Recommendation Systems try to show the most relevant results to the user. We usually make choices based on recommendations from other people who have a similar profile to ours to choose movies, books, etc.

People who dominate a subject can determine whether a document is interesting or not just by quickly browsing through it. However, they can not externalize the rules on which they base their evaluation of a material because this decision relies mainly on tacit knowledge and on the evaluator's personal competences. This kind of solution is especially useful to Knowledge Management whose main challenge is to identify and to enable the right knowledge (tacit and explicit) at the appropriate time.

The proposal of this work is the creation of a Recommendation System which was incorporated into a Scientific Knowledge Management environment, allowing researchers and students to receive, in an optimized way, explicit and useful knowledge for their research process. This recommendations are focused to improve the learning process, reducing the amount of documments read, without reduce the quality of the study. The MISIR approach was implemented into GCC, an environment for KM in scientific research, used by teachers and students. Apart from selectively disseminating explicit knowledge, this proposal also helps in identifying tacit knowledge and competences of the institution.

To describe this work, a small revision of Recommendation Systems (section 2) will be made at first, showing different approaches, as well as advantages and disadvantages of each one. Next, the application scenario for this solution will be shown (section 3), as well as the approach itself (section 4). For a better understanding a small case study will be provided (section 5). Finally we will mention future paths for works and the reach the conclusion of this work (section 6).

2. RECOMMENDATION SYSTEMS

These systems are especially useful when the universe of choices is big and unknown, when the user does not have enough knowledge or expertise to decide, or when there is a record of similar behaviour from other people. There are two approaches to Recommendation Systems (Balabanovic and Shoham, 1997:66-72): the content-based approach (section 2.1) and collaborative approach (section 2.2). It is important to remember that there are hybrid approaches to making a recommendation, which combine the two approaches.

2.1 Content-Based Approach

The premise, in which the content-based approach relies on, is that the user would like to see similar documents from others, which have already been seen and well evaluated previously. In this approach the similarity is calculated among evaluated documents and others which have not yet been evaluated by the user. If the similarity among the evaluated and non-evaluated documents is big, and the evaluated document had obtained a good grade, the non-evaluated document is recommended.

One of the main problems related to this approach is to maintain the recommendation for the subjects that have been specifically evaluated by the user. A document on a new subject would never be recommended because it would not be similar to any of other documents which were positively evaluated by the user (Balabanovic and Shoham, 1997:66-72).

Furthermore, the concept of similarity among documents is the same of quality similarity. So, a document which was recommended by this approach may not necessarily have quality comparable to the document with which it was compared in the recommendation. This means that two documents can be similar, however with very different quality between them. One last factor to bear in mind is that there still are no efficient methods to compare objects such as video and audio (Shardanad and Maes, 1995:210-127) because content analysis is made especially by mining techniques.

2.2 Collaborative Approach

In the collaborative approach the system seeks similarities among the users in order to recommend a well-evaluated document for a similar user. The approach follows the principle that users are seeking a document which has already been evaluated positively by other users similar to him/her (Resnick et al, 1994:175–186). This kind of evaluation type is supposedly more efficient because people evaluate documents in a more proficient way than computational systems, due their previous experiences, expertise, and understanding of the area, quick association with correlated works and comparison with work of the same area.

Systems relying exclusively on content-based filtering recommend only items closely related to those the user has previously rated, while collaborative-only solutions suffer from a cold-start problem, needing a large number of users have rated items in its databases (Salter and Antonopoulos, 2006:35-41).

In the collaborative approach it is possible to solve the problem found in the recommendation made via a content-based approach where the user only receives documents of similar contents making the access to new subjects and the possible analysis of existent subjects more difficult.

3. APPLICATION SCENARIO

With the intention of aiding Knowledge Management in research centers and universities the GCC was created (Oliveira et al, 2005). There are several kinds of explicit scientific knowledge, but mainly scientific

knowledge is disseminated in the shape of documents, as publications, theses and technical reports. Faced with this scenario, a recommendation module was incorporated to GCC.

In the initial page of the system the user is notified about the documents which have not yet been evaluated by him/her but that can be of his/her interest as shown in Figure 1. The user can upload and evaluate a document in his/her community but he/she can navigate to existing communities to access available documents and to evaluate them at any time, as shown in Figure 2.

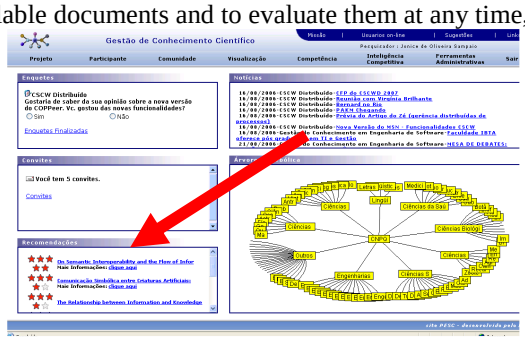

Figure 1. The user is notified about new interesting documents when logging into GCC.

Figure 2. The user can evaluate documents posted in a community.

4. MISIR APPROACH

The MISIR approach aims at facilitating access to a variety of documents for GCC users. To do so the documents must support users' evaluations. With users' evaluations it is possible to predict grades which were not given by other users. When a document has a previewed grade to a user and this grade is above a set minimum limit value, this document will be recommended to the user.

We can divide the MISIR Approach in two stages: a prediction stage and recommendation stage. In the prediction stage, the algorithm seeks documents that the user has not evaluated and, based on each grade given by the other users and it calculates a grade for each non-evaluated document found. In the recommendation stage documents whose predicted grades were higher than a stipulated limit are selected.

In the first stage the similarity among users is calculated taking into account the "users x documents" matrix. Each line of the matrix contains a series of grades which are the U_i user's evaluations for document D_j. If user U_i has not evaluated a certain document D_j all the users that evaluated document D_j are selected. Then, similarities are calculated between each one of the selected users and U_i. After that the prediction grade is calculated starting from each user's evaluation and his/her respective similarity.

In the following stage, all predicted grades are verified and, if the prediction grade is higher than the reference value, the document will be chosen for recommendation to the user.

By analyzing the comparisons for approaches, models and kinds of algorithms which can be developed, related to the reality in which GCC is found, we decided to invest in the collaborative approach, using the model in memory and the Pearson's algorithm's Correlation (Breese et al, 1998:43-52).

This choice proves ideal for our case because the GCC does not simply recommend documents, eliminating the content-based approach. The recommendation based on memory was chosen because it is more effective than the recommendation based on the model, as we can see in tests done by (Breese et al, 1998:43-52), and of easy implementation. In this section we will describe, in general terms, the Pearson's algorithm's Correlation. Initially, for the calculation of grade prediction (Breese et al, 1998:43-52), we used Equation 1.

$$P_{a,j} = m_a + \frac{\sum_i w(a,i) \times (n_{i,j} - m_i)}{\sum_i |w(a,i)|} . \quad (1) \qquad m_i = \frac{\sum_j n_{i,j}}{N_i} . \quad (2)$$

Where

$P_{a,j}$ is the prediction grade of user a for item j;

a is the user who wishes to make the prediction;
j is the document upon which we wish to calculate the evaluation;
w(a, i) is the correlation between user a and user i;
m_i is the arithmetical mean of user i's evaluations;
$n_{i,j}$ is the grade which was given to document j by user i;

Calculation of the arithmetical mean of a user's grades is given by Equation 2.

Where
m_i is the arithmetical mean which we wish to calculate;
j is a document which was already evaluated by the user;
N_i is the number of documents evaluated by user i;
$n_{i,j}$ is the grade which was given to document j by user i;

For the Pearson Method, the value of w(a, i) is calculated based on the users' correlation. The correlation between user *a* and user *i* is calculated by Equation 3, where j is a document which was evaluated by both users.

$$w(a,i) = \frac{\sum_j (n_{a,j} - m_a) \times (n_{i,j} - m_i)}{\sqrt{\sum_j (n_{a,j} - m_a)^2} \times \sqrt{\sum_j (n_{i,j} - m_i)^2}} \quad \textbf{(3)}$$

The Pearson coefficient w(a,i) obtains values in the [-1, +1] interval. The closer it is to -1, the less related the users are. On the other hand, the closer it is to +1, the more similar they are.

4.1 The Workings of Recommendation

We will follow an example of operation of the Pearson's algorithm and give step-by-step example as to how the Recommendation System works.

The matrix shown in Table 1 contains users' grades for documents (those grades vary from 1 to 5). We will carry out grade prediction for Document 3 by User 4.

In the first step, we calculated the averages for users' grades without taking into account the predictions of grades done already or documents which were not evaluated. The result of this example is in Table 2.

Table 1. Evaluation Matrix: Users vs. Documents

	D1	D2	D3	D4	D5
U1	4		5	2	4
U2	3	2		3	3
U3	5	4	4	1	3
U4	2	1		5	2
U5	3	3	1	4	

Table 2. Grade average by users to Document 3

	Average
User 1	3,75
User 2	2,75
User 3	3,4
User 4	2,5
User 5	2,75

We needed to calculate w(User 4, User 1), w(User 4, User 3) and w(User 4, User 5) below. Notice that it is not necessary to calculate w(User 4, User 2) because User 2 did not evaluate the document. Another detail is that we did not need to calculate w(User 4, User 4) because it is easy to conclude that w(User 4, User 4) = 1 because an user is always 100% similar to him/herself.

$$w(4,1) = \frac{(2-2{,}5)\times(4-3{,}75) + (5-2{,}5)\times(2-3{,}75) + (2-2{,}25)\times(4-3{,}75)}{\sqrt{(2-2{,}25)^2 + (5-2{,}5)^2 + (2-2{,}5)^2} \times \sqrt{(4-3{,}75)^2 + (2-3{,}75)^2 + (4-3{,}75)^2}}$$

So:

$$w(4,1) = \frac{-4{,}625}{\sqrt{6{,}75} \times \sqrt{3{,}1875}} \;;\; w(4,1) = \frac{-4{,}625}{4{,}638} \cong -0{,}997$$

This degree of correlation shows that User 1 is almost always contrary to User 4 (see it in Table 1). His/her opinions are always very different and that is the reason why the coefficient is closer to -1.

In repeating the calculations for other users, we reached the following results:

$w(4,3) \cong -0,841$; $w(4,5) \cong -0,683$

With the values of the Pearson's Correlation, we can execute the grade prediction.

$$P_{4,3} = 2,5 + \frac{(-0,997) \times (5 - 3,75) + (-0,841) \times (4 - 3,4) + (0,683) \times (1 - 2,75)}{|(-0,997)| + |(-0,841)| + |(0,683)|}$$

$$P_{4,3} = 2,5 + \frac{(-2,94)}{2,521} \cong 2,5 - 1,166 \cong 1,33$$

Documents are recommended according to a criterion of precision. The higher the threshold, the higher the efficiency of the recommendation will be; however, a smaller number of documents will be recommended. Concluding the example recommendation, we have Table 3 as a result.

Table 3. Matrix of evaluations: Users vs. Documents (after prediction)

	Document 1	Document 2	Document 3	Document 4	Document 5
User 1	4	4,4	5	2	4
User 2	3	2	1,6	3	3
User 3	5	4	4	1	3
User 4	2	1	1,3	5	2
User 5	3	3	1	4	2,6

Should our chosen threshold be 4 then document 2 will be recommended to User 1. In spite of having recommended only 1 document, we should emphasize that this document represents 25% of the documents available for recommendation.

5. CASE STUDY

With the intention of demonstrating the efficiency of the recommendation, a small evaluation of the system will be described. The evaluation consists of comparing grades that the users had evaluated and their respective recommendations that would be made by the system. Comparing these two grades we can prove the efficiency of the recommendation. Thus, 10 books were chosen from the degree course in Computer Science. Five graduation students from the Computer Science course at UFRJ took part in the evaluations. These students used the GCC to evaluate the books. Thus, 50 evaluations of the users were obtained.

The evaluation can be divided into three steps. The first step consists of randomly erasing part of the evaluations. In the second step the algorithm is executed and does the prediction for the erased grades. In the third and final step, the precision of the algorithm through metrics defined in (O'Connor and Herlocker, 1999) is calculated.

The identification codes and the title of books are listed in Table 4.

Table 4. List of different books used in different classes

Student Id	Title (in Portuguese)	Class
1	C Completo e Total	Computing I and Computing II
2	Análise Combinatória e Probabilidade	Combinatory Analysis
3	Números Inteiros e Criptografia RSA	Integer Number and Cryptography
4	Estruturas de Dados e seus Algoritmos	Data Structure I
5	Introdução aos Sistemas Digitais	Logical Circuits
6	Física vol. I – Haliday	Physics I
7	Redes de Computadores e a Internet	Network
8	Estatística Básica	Statistics and Probability
9	Engenharia de Software	Software Engineering
10	Compiladores: princípios, técnicas e ferramentas	Compilers I

The 50 evaluations are represented in Table 5. For the sake of simplification the document's code will be used as its identification in the following tables of this study.

In the first step of the study, we randomly erased 20% of the books evaluations. Notice that they could be different percentages. The erased grades are marked in Table 5. The erased grades were predicted. In this step, during the execution of the prediction algorithm, three matrices were generated: matrix of arithmetical mean for the user's evaluations, matrix of correlation between the users and matrix of prediction.

The first matrix calculated in the algorithm is the arithmetical mean of the user's evaluations, generated for the program and illustrated in Table 6.

Table 5. Material Grades by Users

		Students (Users)				
		André	Daniel	Eduardo	João	Robson
Books	1	3	3	5	5	5
	2	3	2	3	3	4
	3	4	5	5	5	5
	4	3	5	5	5	5
	5	4	3	4	5	4
	6	4	3	3	1	3
	7	4	4	5	3	2
	8	1	4	3	1	1
	9	2	4	4	5	2
	10	3	2	4	4	5

Table 6. Average Grades by Users

Student	Average
André	2,875
Daniel	3,500
Eduardo	4,125
João	3,625
Robson	3,750

The second generated matrix contains the correlations between the users, as shown in Table 7. The correlation between the users is calculated on the basis of the documents that both had evaluated.

Table 7. Symmetric matrix w obtained by Pearson's algorithm

Correlation Coefficient *w(a,i)*		User i				
		André	Daniel	Eduardo	João	Robson
User a	André	1	0,017	0,659	0,218	0,479
	Daniel	0,017	1	0,622	0,306	-0,203
	Eduardo	0,659	0,622	1	0,669	-0,133
	João	0,218	0,306	0,669	1	0,271
	Robson	0,479	-0,203	-0,133	0,271	1

The matrix of correlation between the users is symmetrical. Correlations get values between [-1, +1]. The closer the correlation is to +1, the more similar the users are in the way of evaluating, i. e., they like or dislike the same documents. The closer the correlation is to -1, the more dissimilar the users are in the way of evaluating.

The third matrix calculated in the algorithm is equivalent to the prediction of the erased grades. The results are illustrated in Table 8.

Table 8. Matrix of material's grades by users, with 20% of prediction

Book ID	Students				
	André	Daniel	Eduardo	João	Robson
1	3	3	4,377	5	5
2	3	2	3	2,863	4
3	4	5	5	5	5
4	3	5	5	5	3,761
5	2,900	3	4	3,513	4
6	4	2,300	3,327	1	3
7	2,582	4,113	5	3	2
8	1	4	3	1	2,313
9	2	4	4	5	2
10	3	2	4	4	5

The prediction having been concluded, we are going to measure the precision of the prediction algorithm. Initially, it is necessary to get the error for each prediction, comparing the grades calculated by the algorithm (prediction's grade) with the user's evaluation (real grade).

In (O'Connor and Herlocker, 1999) the author proposed to evaluate the collaborative filtering algorithms on the basis of Mean Absolute Error (MAE), shown in Equation 4. The closer the MAE is to 0, the more precise the algorithm is.

In the same way, we define the Mean Percentage Error (MPE), as in Equation 5.

$$MAE = \frac{\sum_i |E_i|}{N} \quad . \quad \textbf{(4)} \qquad\qquad MPE = \frac{\sum_i |pE_i|}{N} \quad . \quad \textbf{(5)}$$

where:

i corresponds to a prediction;

Ei is the error of each prediction;

pEi is the percentage error of each prediction;

N is the number of predictions carried out;

Indexes MAE and MPE were calculated using the data collected in this study. Predictions' values are listed in Table 9. And, as expected, it was found that the Mean Absolute Error value is close to zero, as demonstrated in (Breese et al, 1998:43-52).

Table 9. List with prediction grades

Student	Book	Prediction	Grade	Difference	% Error	Abs. Error	
André	5	2,900	4	-1,100	22,0	1,1	
	7	2,582	4	-1,418	28,4	1,418	
Daniel	6	2,300	3	-0,700	14,0	0,7	
	7	4,113	4	0,113	2,3	0,113	
Eduardo	1	4,377	5	-0,623	12,5	0,623	
	6	3,327	3	0,327	6,5	0,327	
João	2	2,863	3	-0,137	2,7	0,137	
	5	3,513	5	-1,487	29,7	1,487	
Robson	4	3,761	5	-1,239	24,8	1,239	
	8	2,313	1	1,313	26,3	1,313	
				-4,951	169,2	8,457	Total
				MPE	16,9	0,8457	MAE

Finally, in possession of the predictions, the recommendation system rounds the predicted grade to the next integer and verifies if it reached the threshold that we stipulate as being grade 4 (in a 1 to 5 scale). Thus, the MISiR selects the documents to be recommended for each user, as demonstrated in Table 10 of our example, where 40% of the predictions were converted into recommendations.

We can see that the system makes many recommendations and these recommendations are precise (as it can be found by comparing Tables 5 and 8). It indicates that it is possible to have a good idea of the user's taste only by comparing his/her evaluations with other users' evaluations.

The pure MAE index value can lose some of its meaning since we are not making comparisons of efficiency between recommendation systems. However, its percentage error, Mean Percentage Error (MPE) is of 16.9%. This means that the predicted grades have, on average, 16.9% error.

Another thing that can be seen is that, in their majority, the biggest errors occur with grades 1 and 5. This happens because these limits behave in an asymptotic way in relation to the recommendation. Then, even if these errors are physically bigger on average, they are less representative because they had to be even bigger to interfere in the recommendation of a single document.

Table 10. List of Recommendations

Student	Book	Predicted Grade
Daniel	Redes de Computadores e a Internet	4,113 ≅ 4
Eduardo	C Completo e Total	4,377 ≅ 4
João	Introdução aos Sistemas Digitais	3,513 ≅ 4
Robson	Estruturas de Dados e seus Algoritmos	3,761 ≅ 4

6. CONCLUSION AND FUTURE WORKS

This work presented an environment of recommendation applied in a Scientific Knowledge Management scenario. As seen in the previous section, this environment assists in several stages of the KM process also being a useful tool in the identification of each person's competences in a research centre or university.

Although it has advantages, this approach does not deal with problems such as the insertion of new documents in the database, which will be recommended only later, after some users have evaluated it. A tougher issue to correct is the treatment given to users who have interests that not fit in groups. This particular kind of user won't have similar users which the MISIR can rely on to make the predictions.

The biggest advantage of the chosen approach is its extensibility. The described algorithm can be used to recommend any types of explicit knowledge and widely used in the scientific scenario, such as links, audio, video, processes, mental maps, models, practices, amongst others.

As future work we can mention the enlargement of the proposal aimed at the work with these kinds of explicit knowledge. We have planned an usability study, with real users, to verify if the recommendation is really used by the students. Also, we will extend this proposal to a hybrid proposal, taking into consideration the user's experience and expertise in order to ponder the evaluation provided. For this we will use this approach – which has already been developed in the GCC environment as GCC tools - to identify specialists and measure individual knowledge.

REFERENCES

Balabanovic, M., Shoham, Y., 1997. Fab: Content-Based, Collaborative Recommendation. *Communications of the ACM*, v. 40, n. 3, pp. 66-72.

Breese, J. et al., 1998. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. *Proceedings of Fourteenth Conference on Uncertainty in Artificial Intelligence*, Madison, WI, USA, pp. 43-52.

Herlocker, J. et al., 1999. An algorithmic framework for performing collaborative filtering. *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, Berkeley, California, USA, p.230-237.

O'Connor, M., Herlocker, J., 1999. Clustering Items for Collaborative Filtering. *Workshop on Recommender Systems: Algorithms and Evaluation*, Conference on Research and Development in Information Retrieval, California, USA.

Oliveira, J. et al., 2005. GCC: An Environment for Knowledge Management in Scientific Research and Higher Education Centres. *Proceedings of I-KNOW '05*, Graz, Austria, June, 2005.

Resnick, P. et al, 1994. GroupLens: An open architecture for collaborative filtering of netnews. *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, Chapel Hill, North Carolina, USA, pp. 175 – 186.

Salter, J., Antonopoulos, N., 2006. CinemaScreen Recommender Agent: Combining Collaborative and Content-Based Filtering. *IEEE Intelligent Systems, vol. 21, no. 1,* pp. 35-41.

Shardanad, U., Maes, P., 1995. Social information filtering: Algorithms for automating "word of mouth". *Proceedings of the Conference on Human Factors in Computing Systems – CHI 95*, Denver, USA, pp. 210-127.