Improvement of prediction in collaborative filtering systems

KHARROUBI Sahraoui
Ibn Khaldoun University, Tiaret
Tiaret, ALGERIA
sahraouikharoubi@gmail.com

Dr NOUALI Omar
Algiers, ALGERIA
onouale@cerist.dz

Dr DAHMANI Youcef
Ibn Khaldoun University
Tiaret, ALGERIA
dahmeni_y@yahoo.fr

***Abstract*-- A collaborative filtering system (CFS) makes recommendations to users via the similarity and proximity between the profiles and taking into account their historical valuations. In contrast to most of the CFS, which are based on the approach-based users, we adopt the approach based items to improve the quality of recommendation, this process seems flexible and allowed us to integrate other sources of information while making the calculation mode off-line, and then to improve performance and reduce the inconvenience of the lack evaluation, we used the semantic layer objects such as metadata and semantic relationships between items, finally we explored the technique LSI (Latent Semantic Indexing) to reduce the complexity of algorithm and identify the items most corollas. A set of real MovieLens test has been used for experimental tests.**

***Index Terms*-- collaborative filtering, latent semantic indexing, metadata, recommender system, semantic data.**

I. INTRODUCTION

The volume of information available on Internet not ceasing increasing each day what leads to the problem of information overload. It becomes increasingly necessary to develop tools making to filter this gigantic mass of information, as well as possible to target the answers provided to the users, so that they are closer to their needs and waiting's. Indeed, the phase of search for information is based particularly on the manner of reaching information through requests or by navigation. The situation is currently paradoxical i.e. the hug mass of information and the access to relevant information adapted to the needs user's becomes at the same time difficult and necessary. The problem is not the availability of information but its relevance relative with a context of specific use. A filtration system allows selecting and presenting the only documents interesting a user starting from a dynamic source of information (Internet, E-mail, News…),

This user having a relatively stable center of interest called profile. Among the dominant factors which constitute current stakes in the field of information retrieval, it retains: the volume, the heterogeneity and disparity of information [7].

II. RELATED WORKS

Collaborative filtering exploits the evaluations that users made for certain documents, in order to recommend these same documents to other users, and without it being necessary to analyze the contents of the documents [4].

''Breese'' [4] propose an interesting classification for the techniques of collaborative filtering: memory-based algorithms, model-based algorithms and hybrid algorithms.

A. Memory-based Algorithms

Memory-based Algorithms use the entire database evaluations of the users to make the predictions [3].

If I_i is the set of the items rated by user i, then the average evaluation for user i can be defined as:

$$v_i = \frac{1}{|I_i|}\sum_{j \in I_i} v_{ij}$$

The predicted score on item j for the active user a is a weighted sum of scores of the other users:

$$P_{a,j} = \overline{v_a} + k\sum_{i=1}^{n} w(a,i)\left(v_{ij} - \overline{v_i}\right)$$

Where n is the number of users in the database that have a nonzero weight and k is a normalization factor such as the sum of the absolute values of the weights equal 1.

The weight $w(a,i)$ is determined in different ways, depending on the algorithm.

These algorithms have the advantage of being simple to implement, the predictions are good, as are recalculated every time, but the problem is the high combinatorial complexity $O(m^2 n)$ (m: number of users, n: number of items) [2].

B. Model-based algorithms

Model-based algorithms employ the database user's evaluations to estimate or learn a model which will be useful for calculation of the predictions [14]. From the probabilistic point of view, the task of prediction of an evaluation can be seen like the calculation of the hoped value of an evaluation, being given what one knows of a user. Suppose that the evaluations are done on a scale of integer from 0 to m, and then the predicted value will be:

$$P_{a,j} = E(v_{a,j}) = \sum_{i=0}^{m} P_r(v_{a,j} = i \mid v_{a,k} \in I_a) i$$

Where the expressed probability is that whose active user will make the particular evaluation i for item j taking into account the evaluations observed before [3].

C. Hybrid algorithms

These algorithms combine both collaborative and content-based approaches, one simple approach is to allow both content-based and collaborative filtering methods to produce separate ranked lists of recommendations, and then merge their results to produce a final list [1].

The objective of our study is to make an improvement by various points of seen for these systems CFS which mark some challenges [10], among which:

- *Scalability*

The algorithms used by most memory-based CFS require computations that grow according to the number of users and items.

- *Sparcity*

Following the very high number of items, it is less probable to find evaluations common between users (a minimum of similar neighbors) in this case the comparison is reduced between the active user (target) and the similar neighbors.

- *Could start*

That for a new user (or item) integrated into the system, knowing that the calculation of the similarity depends on the history of the evaluation of the active user and that of the other similar neighbors, in our case its history (profile) is empty, whereas the system provides recommendations can be unsatisfactory for this new user [8].

III. APPROACHES PROPOSED

First, the adoption of item-based approach for collaborative filtering algorithm unlike usual systems that are based on user-based approach (on line mode) to calculate the similarity which implies It calculation of the similarity between items is done in off-line mode (regular time, batch mode, etc.…) that increases system performance.

The space of the items is relatively small compared to the space of the users what shortened the calculation of the correlations between these items.

Secondly the possibility to integrate external data (demographic data, metadata, semantic information, interpersonal relationships between items,) with the system to reduce the problem of unavailability of the users evaluation and reducing the effect of the cold start by the exploitation of a domain ontology;

Finally, we apply a technique LSI (Latent Semantic Indexing), to reduce the space of initial treatment in keeping up the importance of information (decomposition in singular values of matrix SVD).

A. Approach item-based

The intuition behind this approach is that the user is interested in objects which are similar [13], this leads to study the relations between items and classified rather than to seek the similarity between a high number of users (traditional CFS).

1) Calculating the similarity between items

We must first identify all the users who rated the items i_p and i_q (two vectors), then use a measure for calculating the similarity between these two vectors such as the cosine measure:

$$\mathrm{sim}(\mathrm{i_p}, \mathrm{i_q}) = \frac{i_p \bullet i_q}{\sqrt{\|i_p\| * \|i_q\|}}$$

Where • is the scalar product.

Or the measurement of correlation:

$$\mathrm{sim}(\mathrm{i_p}, \mathrm{i_q}) = \frac{\sum_{k=1}^{m} (R_{k,p} - \bar{R}_p)(R_{k,q} - \bar{R}_q)}{\sqrt{\sum_{k=1}^{m} (R_{k,p} - \bar{R}_p)^2 . (R_{k,q} - \bar{R}_q)^2}}$$

Where $k = 1..m$: the list of the users evaluating the items i_p and i_q .

$R_{k,p}$: Value of the evaluation of the user *k* for item *p*.

$\bar{R}_p$: Average of the evaluation of item p.

2) Calculation of the prediction

It selects the most similar items (the K nearest neighbors) for the current item then it generates the prediction value for the item through the evaluations of the current user for K similar items.

The sum of the weights gives us:

$$R_{a,k} = \frac{\sum_{t=1}^{K}\left(R_{a,t}.sim\left(i_k,i_t\right)\right)}{\sum_{t=1}^{K} sim\left(i_k,i_t\right)}$$

$R_{a,t}$: Evaluation value of the current user has on t^ieme item similar.

K: size of the most similar items.

B. Semantic knowledge to optimize CFS

As indicated above, the only criterion for measuring similarity is the value of evaluation, but this really depends on other dimensions of color, shape, age, weight, class, field, focus ... etc, so modeling by implicit measures is useful to improve the accuracy of the system.

1) Motivations of semantic knowledge

The semantic attributes of the items give the implicit reasons of a user to be interested or not by such items that in its turn, allows to the system to make inferences on the basis of this additional source of knowledge and improves the recommendation.

Another advantage for a new item that is not struck by the evaluation, so we made use of semantic information and relationships implicit in the generation of predictions using the semantic similarity;

2) Combination of similarity

To find the total value of similarity between the items, we will take consideration semantic similarity between these items and more value for similarity by rating:

$$SimTot = \alpha SimSem\left(i_p,i_q\right) + (1-\alpha)SimEval\left(i_p,i_q\right)$$

$$0 \le \alpha \le 1$$

$SimTot$: Total similarity.

$SimSem\left(i_p,i_q\right)$: Values extracted from semantic matrix of the similarity calculated by a technique of counting arcs.

$SimEval\left(i_p,i_q\right)$: Values extracted from evaluation matrix of the similarity.

α parameter adjusted according to the experimental results.

If α=0 the approach is purely collaborative.

If α=1 the approach is purely semantic.

The formula of recommendation becomes:

$$R_{a,k} = \frac{\sum_{t=1}^{K}\left(R_{a,t}.SimTot\left(i_k,i_t\right)\right)}{\sum_{t=1}^{K} SimTot\left(i_k,i_t\right)}$$

The integration of the semantic data brings two advantages for the CFS, firstly the use of this semantic information improve measurements of similarity and the comparison between objects what increases the precision of recommendation (main interest for these systems).

Secondly, these metadata describe new items integrated into the system what reduces the effect of cold start.

C. LSI Technique

Latent Semantic Indexing LSI, it is an algebraic model of Information Retrieval IR, based on the decomposition in singular values of matrix (term-document) which represents the space vector indexing model [5], this matrix is projected in a lower space of dimension. Many applications IR showed that the application of this technique improves quality of precision.

Here, we apply this idea to create a space of dimension reduced for items matrix (item-attribute), and on the evaluation matrix (user-item) while keeping the importance of information.

Let $Sn \times d$ the semantic matrix of n items and d attributes, by decomposition SVD:

$$Sn \times d = Un \times r \bullet \Sigma r \times r \bullet Vr \times d$$

Where U and V are two orthogonal matrices (U.U '= I), r is the rank of matrix S and Σ is a diagonal matrix of size r × r, where the diagonal contains all singular values of matrix S stored in a descending order.

It is proven that there exists only decomposition in this manner [6].

We can reduce the rank of the diagonal matrix Σ to a lower rank k (k <r) to maintain the k largest singular values, therefore we reduce U to U' and V to V' the matrix approximate S becomes:

$$S'n \times k = U'n \times k \bullet \Sigma' k \times k \bullet V'k \times d$$

D. Provision of semantic web for information filtering

The semantic web and the use of ontology offer major advances:

- Increase in the relevance via the description of the resources by the metadata.

- Each entity being represented using an ontology what facilitates the automation of the tasks.
- Interoperability and automation of tasks alleviates the arduous viewing by the user and saves considerable time.

IV. EXPERIMENTS AND RESULTS

A. *Dataset MovieLens*[1]

The data concern 943 registered, 1682 titles of films and 100000 evaluations, the films are rated on a scale of 1 to 5, and each user has rated at least 20 films (between 20 and 737), total 93.7% of notes are missing (sparse matrix), these data files are very much used for many studies in the field of collaborative filtering [16].

B. *Evaluation process*

First, we implemented the item based algorithm of collaborative filtering on a test set (about 30% of the total base) In the second step, we exploited the movie table for the semantic information; The framework has been created with the Java (eclips SDK3.5) and Matlab.

Finally and in order to optimize the recommendation system we applied a technique LSI based on singular value decomposition of the original matrix and see the results of each paradigm based items, semantic and combined.

C. *Evaluation metrics*

1) *MAE*

Mean Absolute Error [9], calculates the average difference between the predictions p_j calculated by the system and the scores ej really given by the user in the process of the evaluation.

$$\overline{|E|} = \frac{\sum_{i=1}^{N} |e_i - p_i|}{N}$$

N: the number of the items evaluated by the user.

The objective is to minimize this error.

2) *Recall*

The recall is the proportion of relevant items returned by the algorithm compared to the total number of existing relevant items. Its formula is:

$$R = \frac{N_{pr}}{N_p}$$

The recall measures the effectiveness of an algorithm.

3) *Precision*

Precision is the proportion of relevant items among all of those returned by the algorithm.

$$P = \frac{N_{pr}}{N_r}$$

The increase in the value of precision reduces the value of noise and improves the quality of result.

D. *Results*

1) *Algorithm based evaluation*

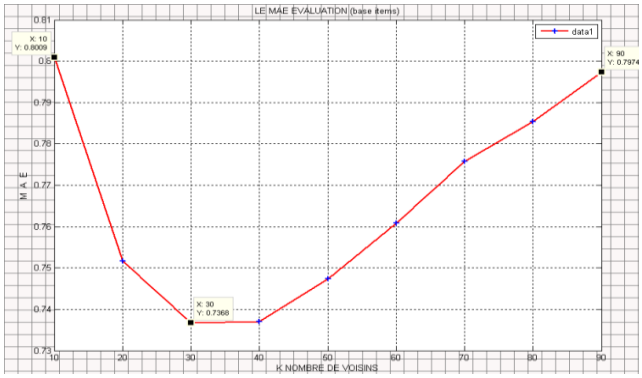

Fig. 1. MAE based evaluation

Fig.1 shows clearly that the MAE passes from 0.8009 per 10 neighbors to the optimal value 0.7368 near 30 to 40 neighbors, then the error increases proportionally with the increase in the number of neighbors what translates logically that the similarity is degraded between items starting from a given row (>40 neighbors) and consequently the automatic increase in error.

2) *Semantic algorithm*

Initial matrix: $ES_{500\times20}$. (500 items, 20 attributes).

Neighborhood size: 10 to 100.

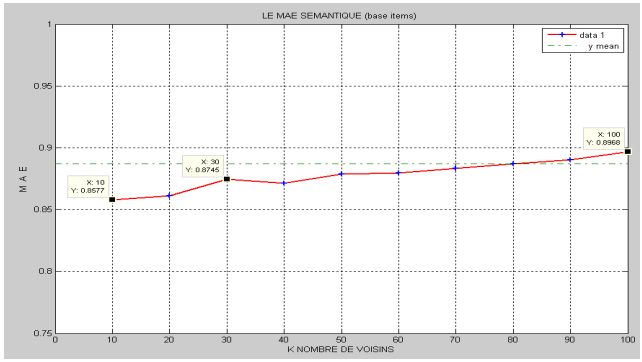

Fig. 2. Semantic algorithm

According to the fig. 5, we note that the error is tolerated by the semantic algorithm.

[1] http://www.grouplens.org/node/12

TABLE I. COMPARISON EVALUATION AND SEMANTIC RESULT

	MAE	Recall	Precision
Evaluation algorithm	0.7368 - 0.8009	8.1%	72.5%
Semantic algorithm	0.8577 - 08968	16.22%	43.1%

3) Hybrid algorithm (evaluation and semantic)

Initial matrix: $E_{200\times400}$, $ES_{500\times20}$.

Neighborhood size: 10 to 90.

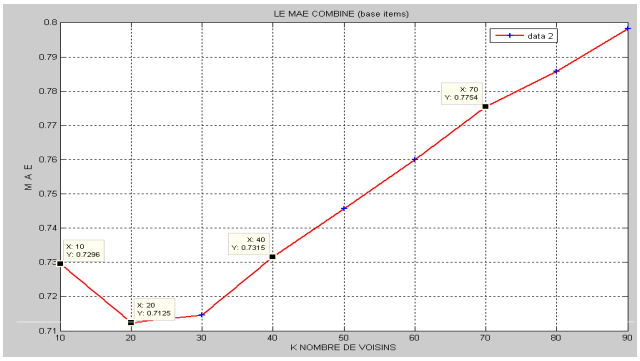

Fig. 3. Hybrid algorithm

The combination of the two algorithms improves the result, a MAE = 0.7125 for the 20 most similar neighbors.

4) LSI

a) Algorithm based evaluation

Initial matrix: $E_{200\times400}$ (200 users and 400 items)

Neighborhood size: 10 to 200.

SVDk: k=10.

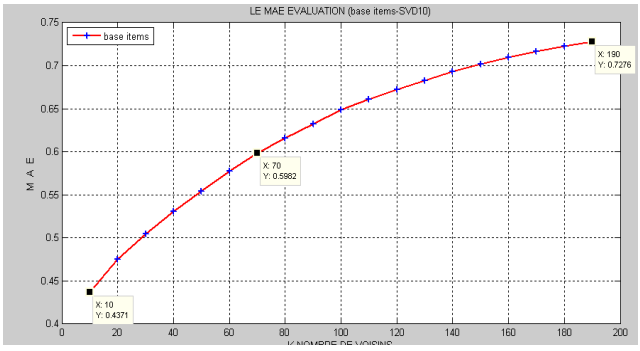

Fig. 4. Evaluation algorithm (SVD10)

By applying the LSI technique showed a remarkable improvement (MAE between 0.4371 and 0.7276).

b) Semantic algorithm

Initial matrix: $ES_{500\times20}$. (500 items, 20 attributes)

Neighborhood size: 10 to 190.

SVDk: k=10

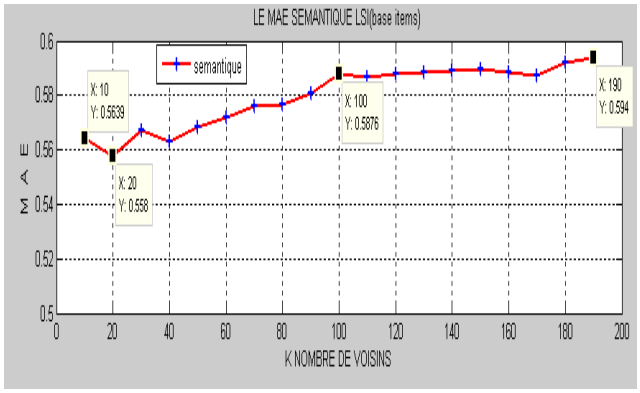

Fig. 5. Semantic algorithm (SVD10)

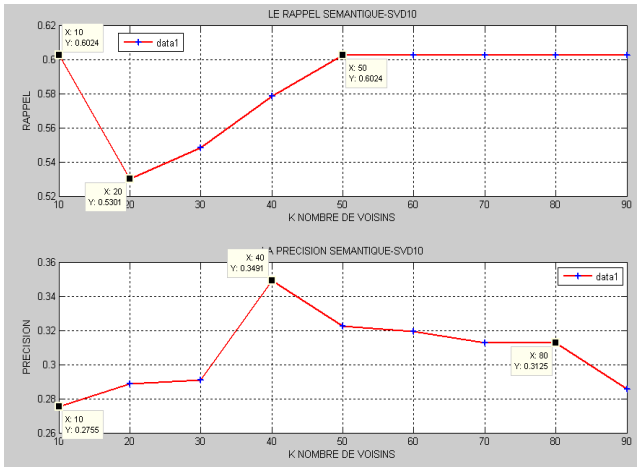

Fig. 6. Semantic algorithm (recall, precision)

It is noted that the application of technique LSI makes an important improvement on all measurements, the mean absolute error, the recall and the precision.

c) Hybrid algorithm (evaluation and semantic)

Initial matrix: $ES_{500\times20}$, $E_{200\times400}$.

Neighborhood size: 10 to 190.

SVDk: k=10.

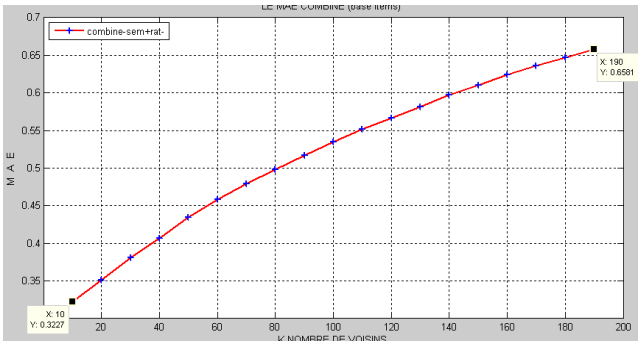

Fig. 7. Hybrid algorithm (SVD10)

The combination of the evaluation users and the semantics of the items gave us good performances; the introduction of technique LSI still improved these results.

TABLE II RESULTED SUMMARY

ALGORITHMS	M A E	Recall	Precision
Evaluation	0.7368-0.8009	4.94%-8.14%	15.19%-72.55%
Semantic	0.8577-0.8968	12.9%-16.22%	33.33%-43.10%
Hybrid	0.7125-0.7985	13.98%-18.25%	35.97%-86.84%
Evaluation by SVD	0.4371-0.7276	30.43%-82.48%	27.70%-46.08%
Semantic by SVD	0.5639-0.5580	53.01%-60.24%	27.55%-34.91%
Hybrid by SVD	0.3227-0.6581	14.65%-94.33%	21.41%-68.27%

The table shows that the addition of semantic information and the introduction of LSI technique; thus a hybridization between these approaches led to satisfactory results (MAE=0.3227) and consequently an improvement of the quality of CFS.

IV. CONCLUSION AND FUTURE WORK

The approach suggested is based on:

Integration of semantic information has allowed us to increase the similarity measure, based on this mass of information in case of absence of the usual information explicit or combining these two types of information to achieve better prediction; initially the filter system can use this information to the recommendation and reduce the effect of cold start.

In order to optimize the algorithm of prediction we applied LSI technique to make the matrices less empty and to reduce the dimension of initial space, this solution gives satisfactory results.

The experimental results prove that the approach based items provided with the semantic infrastructure increases and improves the information recommended while treating the effect of cold start and the problem of the lack of the evaluation. We envisage the effective and optimal exploitation semantic data holding of account the structure of the domain ontology to improve the relevance of information recommended.

It seems useful to study the parameter of combination between semantic information and the information explained by the users according to the requirements and information available.

The integration of the intelligent agents for the treatment of the base profile and the data processing external (corpus) increases the automation and the speed of the process of filtering.

V. REFERENCES

[1] Charu C. Aggarwal, Joel L. Wolf, Kun-Lung Wu, Philip S. Yu, "Horting Hatches an Egg: A New Graph-Theoretic Approach to Collaborative Filtering", 2005.

[2] A. Belloui, O. Nouali ''L'usage des concepts du web sémantique dans le filtrage d'information collaborative'', these de magistère INI-Alger 2008.

[3] C. Berrut, "Filtrage collaboratif" E. Gaussier, M.H. Stefanini, "Assistance intelligente à la recherche d'information", Hermes-Lavoisier, 2003.

[4] Breese J . S, Heckerman D., Kadie C., ''Empirical analysis of predictive algorithms for collaborative filtering, proceedings of the 14th Conference on Uncertainty in Artificial Intelligence'', (UAI'98), Wisconsin, USA, 1998.

[5] S. Deerwester, S.T Dumais, G.W. Furnas, T. K. Landauer, et R. Hrashman. ''Indexing by latent semantic analysis''. Journal of th american society for information science, 1990.

[6] G. Golub et C. Van Loan, Johns Hopkins , Baltimore, ''Matrix computations'',1996.

[7] Lynda Lechani Tamine , Mohand Boughanem, ''Accès personnalisé a l'information Approches et techniques'' ,IRIT 2007.

[8] P. Melville, R.J Mooney , R.Nagarajan, ''Content-boosted collaborative filtering for improved recommendations''. Proceedings of the 18th National Conference on Artificial Intelligence , pp. 187-192, 2006.

[9] B. Mobasher, X. Jin, Y. Zhou, ''Collaborative filtering on the web '', 2004.

[10] B. Sarwar, M. Karypis, Konstan, & Riedl, ''Collaborative filtering recommendation algorithms'' (2004).

[11] J. L. Herlocker, J.-A. Konstan, J.Riedl, ''Explaining Collaborative Filtering Recommendations'', Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work (CSCW'00), Pennsylvania, USA, p. 241-250, 2000.

[12] B. Thành Lê, ''Construction d'un Web sémantique multi-points de vue'' These de doctorat L'École des Mines de Sophia Antipolis octobre 2006.

[13] Z. Zheng, H.Ma, R. Michael, "QoS-Aware Web Service Recommendation by Collaborative Filtering", Published by the IEEE Computer Society 1939-1374/11/$26.00 _ 2011 IEEE.

[14] Z. Fuzhi, S. FENG, J.Dongyan, T. Qing "DCFKQ : A DHT-based Distributed Collaborative Filtering Algorithm"Journal of Computational Information Systems6:1(2010).