ISSN
Number 2 | February 2013

COMPUTATIONAL INTELLIGENCE SERIES

Hybrid Recommendation System based on Collaborative Filtering and Fuzzy Numbers

Miguel A. G. Pinto

Ricardo Tanscheit

Marley Vellasco

Laboratório
Inteligência Computacional Aplicada

COMPUTATIONAL INTELLIGENCE SERIES

Number 2 | February 2013

Hybrid Recommendation System based on Collaborative Filtering and Fuzzy Numbers

Miguel A. G. Pinto

Ricardo Tanscheit

Marley Vellasco

CREDITS
Publisher:
MAXWELL / LAMBDA-DEE
Sistema Maxwell / Laboratório de Automação de Museus, Bibliotecas Digitais e Arquivos
http://www.maxwell.lambda.ele.puc-rio.br/
Organizer:
Marley Maria Bernardes Rebuzzi Vellasco
Cover:
Ana Cristina Costa Ribeiro

Hybrid Recommendation System based on Collaborative Filtering and Fuzzy Numbers

Miguel A. G. Pinto, Ricardo Tanscheit, Marley Vellasco
Department of Electrical Engineering
Pontifical Catholic University of Rio de Janeiro
Rio de Janeiro - Brazil
miguel.thinktank@gmail.com, ricardo@ele.puc-rio.br, marley@ele.puc-rio.br

***Abstract*—Online retail stores face great challenges to recommend products due to the size and sparsity of the databases, as well as the variety of new users and items. As current techniques, based on collaborative filtering, address those issues with only partial success, the present paper proposes the use of a hybrid system of recommendation in online stores. This system makes use of collaborative filtering and of a fuzzy number model based on marketing concepts. Experimental results show that the proposed system presents great invariance to sparse databases, which is of great value for retail companies.**

Keywords-recommendation; marketing; collaborative filtering; fuzzy numbers.

I. INTRODUCTION

Competition between companies, which used to be a local issue, has now become a much larger one due to the widespread access to the internet. A virtual store can now display a large variety of products and reach a much larger number of potential customers.

In the internet early days, companies could prosper more easily, but now they must compete with virtual giants for market slices. Due to the well known fact that online customers are usually impatient, virtual stores normally display the best offer in their webpage to prevent the customer from leaving it and moving to a competitor's.

Offering the right product is the key issue to sales success. Collaborative filtering [1][2] consists of agents that employ the user's behavior and preferences to filter alternatives and make recommendations [3] of items in virtual retail. Such systems are not effective when dealing with large and sparse databases, which is the usual case in commercial applications, or to make recommendation for new users [4], whose purchase habits the system has little knowledge about.

Besides collaborative filtering, content-based filtering is another important class of recommender [5]. A content-based algorithm employs textual information to make recommendations. The text can be obtained from several sources: documents, URLs, news, website logs, description of users and items, user preferences, etc. The recommenders search for patterns in these texts that allow recommendation [6]. While these algorithms can perform well with sparse databases and new users, the amount of information they require may not be available.

This paper proposes a new recommendation algorithm that brings together marketing concepts (product positioning) and successful technologies in recommendation (collaborative filters and content-based algorithms). The resulting system is capable of making recommendations to new users and of performing well with sparse databases.

II. COLLABORATIVE FILTERING

Several companies in the retail sector invest significantly on customers and purchase databases. In virtual stores, such investment is even more important, since those databases are a crucial part of their operations. Besides, algorithms have been developed to make use of such volume of data and generate results. Collaborative Filtering (CF) [7][8] proved to be successful in several applications by searching for similarities in user habits to predict their future decisions.

In Collaborative Filtering, information or patterns are filtered by the use of techniques that involve the collaboration of multiple agents, points of view, sources of information, etc. Collaborative Filters work by building a database to discover a user's neighbors – other users with similar characteristics. The items of interest to those neighbors are recommended to the original user.

Collaborative Filtering is based on the premise that, if two users X and Y have similar interests, reflected on similar evaluations of n *item*s, they shall show the same similarity of interests with respect to other items [9]. A Collaborative Filter (CF) can acquire user opinions explicitly, such as evaluations of some items, or in an implicit way, by using the user's purchase history [10]. The algorithm's objective is to suggest new items or to predict the usefulness of a certain item to a particular user based on his previous preferences or on the preferences of similar users. As a CF typical scenario, let's consider a list of m users $U = \{u_1, u_2, ..., u_m\}$ and a list of n items $I = \{i_1, i_2, ..., i_n\}$. Each user u_i has a list of items I_{ui} that he has expressed interest on. It should be noted that $I_{ui} \subset I$ may be an empty set. The CF's task is to find an item of interest to a particular user $u_a \in U$, named active user. There will be a list of N items, $I_r \subset I$, which will be of more interest to the active user. The recommended list must have items that were not evaluated by the active user, that is, $I_r \cap I_{ua} = \Phi$. This CF interface is also known as Top-N recommendation.

Most CF algorithms are memory-based and use the entire user-item database to generate the prediction. Such algorithms, also known as nearest-neighbors, are vastly used in actual situations, including commercial applications such as Amazon and Barnes & Noble, due to its ease of implementation and high efficacy [11][12]. A particular memory-based method, called item-based, takes the set of items rated by the user and computes how similar they are to a target item. The k most similar items $\{i_1, i_2, i_3,... i_k\}$ are then selected. The corresponding similarities $\{s_1, s_2, s_3,... s_k\}$ are also computed. The prediction is then obtained by considering the weighted average of the user's ratings of similar items [13].

The critical point in an item-based CF is the computation of the similarity between items and the selection of similar ones. The basic idea in the calculation of similarity between two items i and j is, initially, to isolate users who have rated such items and then determine the similarity s_{ij} between them. Fig. 1 shows such process, where the matrix lines represent users and the columns represent items.

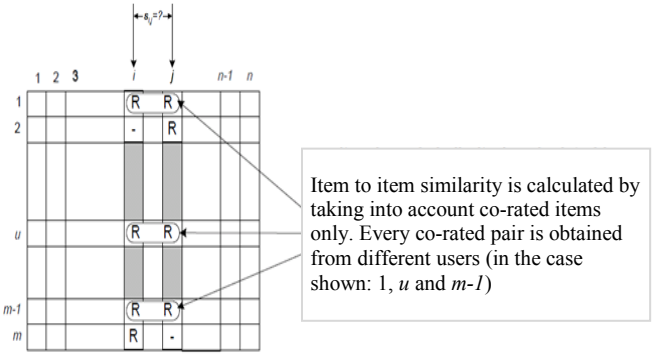

Figure 1. Isolating co-rated items for the computation of similarity

In order to compare items and find out which ones are closest to each other, the item-based CF algorithm employs the cosine similarity of the evaluation vectors of two items for which the similarity will be computed. Assuming that A is an m x n user-item evaluation matrix, the vectors $i=\{i_1, i_2,..., i_m\}$ and $j=\{j_1, j_2,..., j_m\}$, corresponding to the ith and jth column of matrix A, contain the evaluations of items i and j by the m users. The similarity between both items is defined by the cosine between the vectors:

$$w_{i,j} = cos(\vec{i},\vec{j}) = \frac{\vec{i}\cdot\vec{j}}{\|\vec{i}\|\|\vec{j}\|} \frac{i_1 j_1 + i_2 j_2 + \cdots + i_m j_m}{\sqrt{{i_1}^2 + {i_2}^2 + \cdots + {i_m}^2}\sqrt{{j_1}^2 + {j_2}^2 + \cdots + {j_m}^2}} \quad (1)$$

where "·" denotes the dot-product of the two vectors [14].

Collaborative Filtering has been successfully employed in fields such as *e-commerce* and information filtering [5], but there are still important challenges regarding its use in recommendation. In several real applications, databases are too large and sparse, with a large amount and diversity of both users and items, so that a single user will hardly evaluate a substantial quantity of items. Recommenders are also required to show great speed and accuracy in order to bring profit to companies. The main challenges can be summarized as: data sparsity and scale, synonymy (items with the same name in the database) and gray sheep (new user or item, not rated during the CF training).

III. MARKETING AND FILTERING BY FUZZY NUMBERS

When companies develop a new product (or item), they employ marketing techniques to define its specifications. Such techniques are meant to assign value to a specific market niche where the company positions itself. Positioning products requires knowledge of the interests and preferences of a specific group of consumers and capacity of translating this information into numbers, by adjusting value aspects to the maximum value consumers are prepared to pay for a product.

The marketing process for product specification is a three-stage process (Fig. 2): segmentation, selection and positioning. The types of existing consumers are initially determined and then the niche in which the company has better conditions to act is selected. Once the niche is selected amongst the segmented population, the company starts the process of positioning products and services for that selected niche by choosing specifications to better satisfy the costumers of such niche and let the company be distinguished from others in the market. In the computer market, for example, many niches may be identified. Some customers may require a computer for professional reasons, while others may want one only for entertainment. A company will look at this huge market and choose one specific niche, as, for example, people who like to play games. Once the niche is established, the company will position their products that comply with the specifications for that niche. In the case of games, specifications could be a strong graphical card, sound and a fast processor.

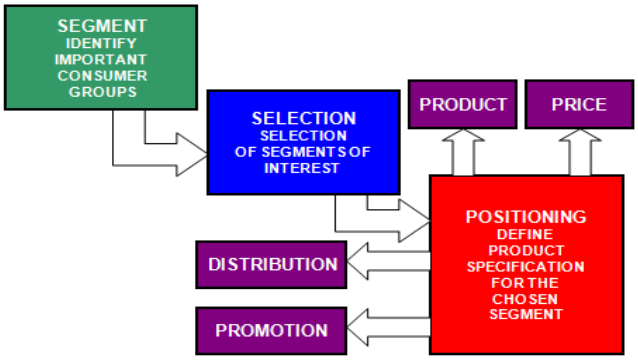

Figure 2. Segmentation, Selection and Positioning

Collaborative Filtering ignores the process that goes from the positioning of a product to the definition of its individual specifications. As collaborative filtering takes into account only the proximity of items or users, it may not perform well when there are insufficient evaluations of items.

Content-based recommenders employ heuristics and classification algorithms to make recommendations [15]. Similarly to what occurs in collaborative filtering, content-based techniques also have an initialization problem. While in the latter the problem lies on the lack of evaluation of an item by a new user (or a poorly evaluated new item), in content-based techniques the problem occurs when there is little information to be analyzed. Moreover, these recommenders are limited to the characteristics explicitly associated to the objects

they recommend, so that experts are often required to complete the information.

The content-based recommender proposed by Hosseinpour [16] has a direct relation to marketing concepts and is of easy hybridization with a collaborative filter. Fuzzy numbers and product characteristics constitute the basis of the recommendation strategy. It must be said that fuzzy numbers have been successfully used in content-based algorithms for web-textual information retrieval [17][18].

The triangular fuzzy numbers used here can be seen as possibility distributions and are denoted by $\widetilde{p} = (p_1, p_2, p_3)$, with membership functions $\mu_{\widetilde{p}}(x)$, where p_1, p_2 and p_3 are real numbers such as $p_1 \le p_2 \le p_3$.

Every item to be recommended can be defined by a set of technical specifications that distinguish it from others in the same category. In the proposed methodology, such technical specifications can be translated into components that have some value to the users. For the evaluation of such components, seven fuzzy numbers are defined, as shown in Fig. 3. The linguistic terms that may be associated to them are very low (VL), low (L), medium low (ML), medium (M), medium high (MH), high (H) and very high (VH).

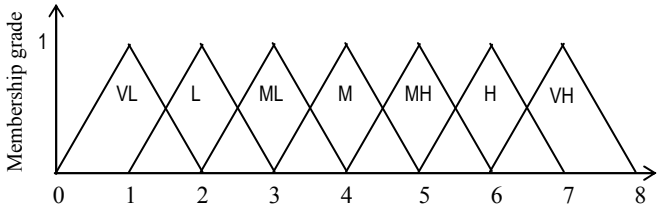

Figure 3. Membership functions for triangular fuzzy numbers

Each item I_i is represented by a vector $\widetilde{P}_i$ with n components related to that item. The same item I_i is also associated to a vector E_i, composed of the technical specifications that distinguish this item from others in the same category. Each component $p_i \in \widetilde{P}_i$ is composed by a vector of specifications $\widetilde{E}_i = (\widetilde{e}_i^{\,1}, \widetilde{e}_i^{\,2}, ..., \widetilde{e}_i^{\,k})$, where each technical specification $\widetilde{e}_i^{\,j}$ is a triangular fuzzy number that represents how that specification affects the components. As each specification can affect the components in distinct ways, a vector of weights $W=(w_1,w_2,...,w_k)$ is considered. It is then possible to calculate the value of a component, as a triangular fuzzy number, from the specifications:

$$p_i = \sum_{j=1}^{k} (e_i^j \times w_i^j) \qquad (2)$$

The component vector $\widetilde{P}_i = (\widetilde{p}_1, \widetilde{p}_2, ..., \widetilde{p}_n)$ is formed by triangular fuzzy numbers where each of them represents a product component and can be used for comparison of user interests.

In the Hosseinpour method, the user defines the set of components he wishes to see in the recommended item, where each component is a fuzzy number, as described previously. All items have its components defined by fuzzy numbers from the specifications, so that they can be compared to the set of components defined by the user.

Suppose a user defines the component for an item of his interest as $\widetilde{q}_B = (q_B^1, q_B^2, q_B^3)$ and that there is an item whose value for the same component is given by the fuzzy number $\widetilde{q}_A = (q_A^1, q_A^2, q_A^3)$. The similarity between those two fuzzy numbers is then given by the near compactness between them:

$$N_E(\widetilde{q}_A, \widetilde{q}_B) = 1 - \frac{1}{\sqrt{3}} \left(\sum_{j=1}^{3} \left| q_A^j - q_B^j \right|^2 \right)^{1/2} \qquad (3)$$

Thus it is possible to make a comparison of the customer's interests in products available for sale by using the value each product offers by way of its components. The lower the value of $N_E(.,.)$, the closer is the analyzed item to the customer's interests.

IV. HYBRIDIZATION

The algorithm described in the previous section resembles the positioning process used in marketing if each component is considered as a product position. Therefore it is possible to generate positions based on specifications and consider that similar users choose items with similar positions. It is also possible to make direct comparisons between items based on their positions.

The main problem of memory-based collaborative filtering algorithms is their loss of performance due to sparse datasets and to recommendations to new users. The recommendation method by fuzzy positioning, on the other hand, has its own disadvantages: the need for an expert to generate positions based on the item specifications, the impossibility of making cross-recommendation between items of different categories, and the need for a user to define positions prior to the recommendation.

The fuzzy positioning recommendation algorithm (simply called "Hosseinpour method" hereafter) does not allow recommendation of items of distinct categories, whereas in collaborative filtering this problem does not arise. Therefore, the use of collaborative filtering as a first stage in a recommendation algorithm seems to be a good approach. The successful hybridization of recommendation algorithms, albeit in other context [19], is an indication that a similar approach could be useful here.

It is proposed here that both algorithms are used in cascade: the collaborative filter initially chooses the proper category and then the fuzzy recommender takes the final decision on which items of the category should be presented to the user. In more specific terms, the item-based CF defines which category will be recommended. As this filter defines categories and not items, it will be named "Category-based CF". Thereafter the fuzzy positioning algorithm recommends a top-N set of items within the category.

In the algorithm based on fuzzy numbers proposed by Hosseinpour the user has to define positions manually, which is unpractical for commercial recommenders. Therefore, if recommendations are based only on the user's purchase history, fuzzy positions of purchased items can be used to infer the user position and allow the similarity calculation by considering the items at disposal for recommendation. The final algorithm proposed here is called "Hybrid Category-Fuzzy CF".

V. EXPERIMENTS

To evaluate the proposed algorithm an experimental database was created, with data representative of the Brazilian *e-commerce*. Categories, positions and purchase rules were created to simulate a real-world database and allow tests of sparsity.

The experiments were undertaken by considering 20,000 clients representative of the characteristics of the Brazilian society obtained from the demographic census. Fourteen categories were defined and technical specifications and their respective positions were assigned to each of them. In total, the database had 1,000 items for each category, with technical specifications created at random, originating a vast quantity of specifications and, therefore, positions. By taking into account average purchases of Brazilians in the e-commerce, seven purchases per user have been considered. Purchase rules considered users' characteristics (sex, age, wealth, etc.) and positions were created for each user. These positions were compared to the ones originated from items specifications by using the Hosseinpour method. Thus, a relation between each user and each rated item in the database was established. Through this method, a given user could evaluate all items and sparsity could be varied freely by increasing or reducing the number of users with respect to the number of available items. In a synthetic data base, as is the case here, all items can be evaluated, since there is complete information about all users and products specifications. In the experimental stage, evaluations were was performed offline due to the long processing time.

It may be argued that the use of the Hosseinpour method to create purchases in the synthetic database could affect performance positively. However, in the experiments, users' specifications were not taken into account. Each product was evaluated by comparing its position to those of other products acquired by the user. Besides, the purpose was mainly to evaluate the algorithm's behavior with respect to sparsity. In an ongoing work with a real database, performance is being considered.

The item-based CF and the Category-based CF were trained over the purchase database, by using the cosine similarity and considering different sparsities. The items were compared by using the fuzzy positions and the near compactness as the similarity measure. The sparsest database contained 5,000 users and the least sparse considered 20,000 users.

In order to evaluate the top-N algorithms, the metrics of *precision* and *recall* were used. The training set (*CTr*) consisted of the first six user's purchases, while the last one formed the test set (*CTe*). After training, a set of recommendations was created (top-N), which size may change. Eventually, the items that belong both to top-N and *CTe* form the target set. The metrics are:

$$recall = \mathrm{Re} = \frac{|CTe \cap top-N|}{|Cte|} \quad (4)$$

$$precision = \mathrm{Pr} = \frac{|CTe \cap top-N|}{N} \quad (5)$$

The final evaluation is given by:

$$F1 = \frac{2 * \mathrm{Pr} * \mathrm{Re}}{\mathrm{Pr} + \mathrm{Re}}$$

Figs. 4 and 5 show the values of F1 for several levels of sparsity and several values of Top-N. Each graphic shows the results of the algorithms on databases with an increasing amount of users, causing sparsity to decrease. The Hybrid Category-Fuzzy CF performs significantly better than the item-based CF, especially for smaller numbers of N. Due to the importance users give to the first presented items, such characteristic has extreme significance.

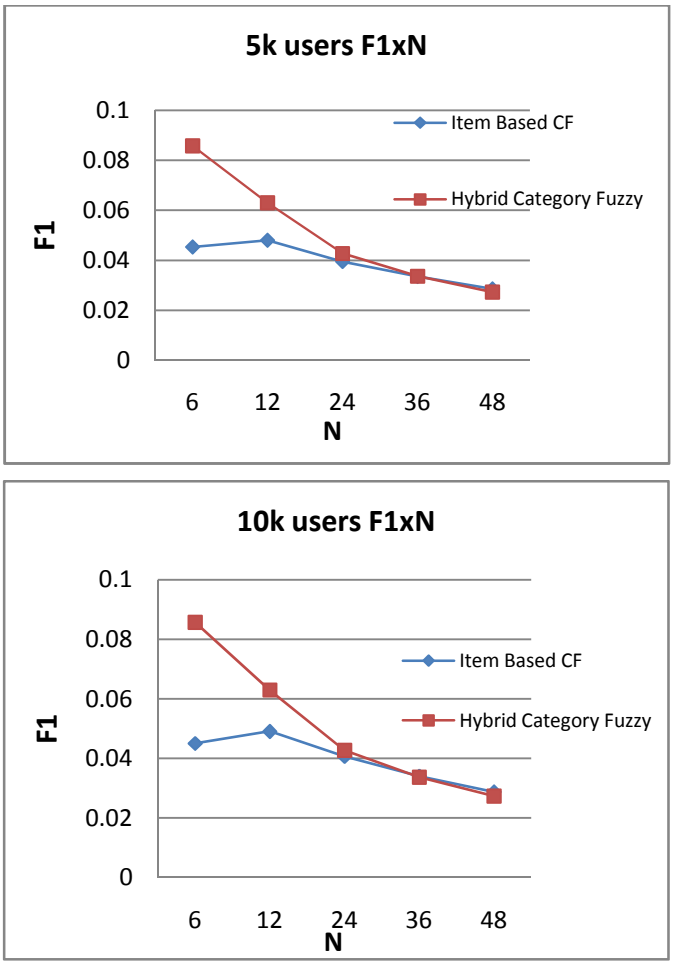

Figure 4. Experiments for several values of top-N and training for several levels of sparsity

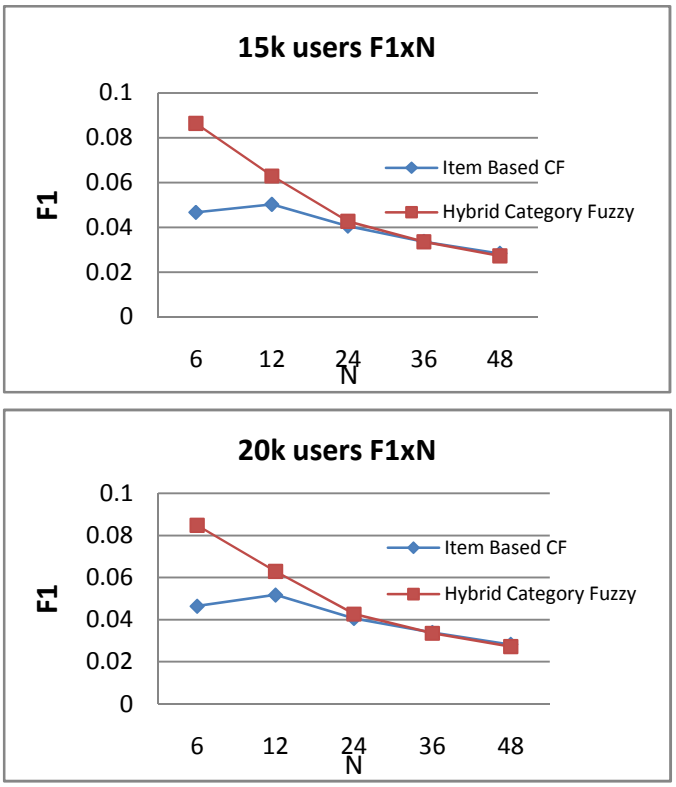

Figure 5. Experiments for several values of top-N and training for several levels of sparsity.

Another aspect of the Hybrid Category-Fuzzy CF with relation to the item-based CF is the fact that the former is practically not affected by changes in the sparsity of the database; this is not the case for the Item-base CF. This conclusion is reached by observing the value for F1 in each case. This characteristic is more evident in Fig. 6, which shows the results (F1) for both algorithms against the number of users. The invariance to sparsity is a very important factor in real databases.

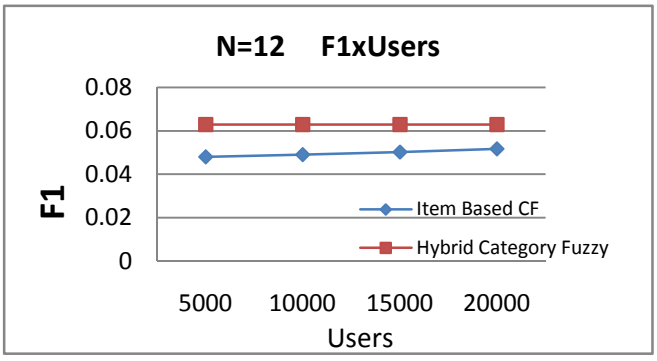

Figure 6. Experiment for several values of users, keeping the number of recommendations at 6.

Fig. 7 shows the behaviors of the item-based CF, the Hybrid Category-Fuzzy CF, the Category CF and the Hosseinpour method (by considering that the category is always right) for several values of Top-N and by varying the number of purchases of each user.

By changing the number of user's purchases it is possible to analyze the case of a "black sheep" (a new user makes the first purchase and the CF has only that purchase to make the recommendation).

The performances of all algorithms improve as the number of purchases increases. Both the Hosseinpour method and the Category- based CF perform better than the Item Based CF and the Hybrid Category-Fuzzy algorithm. Despite being composed of two successful algorithms, the Hybrid Category-Fuzzy loses some performance because it only reaches a recommendation when both constituting algorithms do so. The Hosseinpour method shows a superior performance when compared to the item-based CF in all cases, and the Hybrid Category-Fuzzy CF performs closely to the Hosseinpour Method when the Category-based CF increases its performance.

VI. Conclusions

A new recommendation algorithm was developed that explores the interaction between current recommendation technologies (content-based and collaborative filters) and concepts of marketing. The results obtained justify the proposal.

The Category-based CF offered to the Hosseinpour method (content-based) the possibility of recommending items from distinct categories, while the Hosseinpour method allowed a higher precision on recommendation of items inside the same category. As predicted, hybridization compensated for the weaknesses of the individual algorithms.

The model proposed here combined fuzzy numbers, product positioning (from marketing theory) and item-based collaborative filtering. The resulting system showed a better performance when compared to the item-based CF, generating satisfactory results for a diversity of recommended items. The proposed filter has also shown great invariance to sparse databases, which is of great value for retail companies. It has also been shown that an experimental database is viable for evaluating recommendation systems; in further works real databases will be used. Collaborative filters and Hosseinpour method were tested with success in real cases; therefore equally satisfactory results, now evaluating performance, can be expected.

References

[1] P. Resnick and H.R. Varian, "Recommender systems," Communications of the ACM," vol. 40, no. 3, pp. 56–58, March 1997.

[2] J. B. Schafer, J.A. Konstan, and J. Riedl, "E-commerce recommendation applications," Data Mining and Knowledge Discovery, vol. 5, pp. 115–153, January 2001.

[3] A. Ansari, S. Essegaier, and R. Kohli, "Internet recommendation systems", J. Marketing Research, vol. 37, pp. 363–375, August 2000.

[4] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions," IEEE Transanctions on Knowledge and Data Engineering, vol. 17, pp. 734–749, June 2005.

[5] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," Advances in Artificial Intelligence, pp. 19, 2009.

[6] G. Karypis, "Evaluation of item-based top-N recommendation algorithms," International Conference on Information and Knowledge Management , pp. 247–254, 2001.

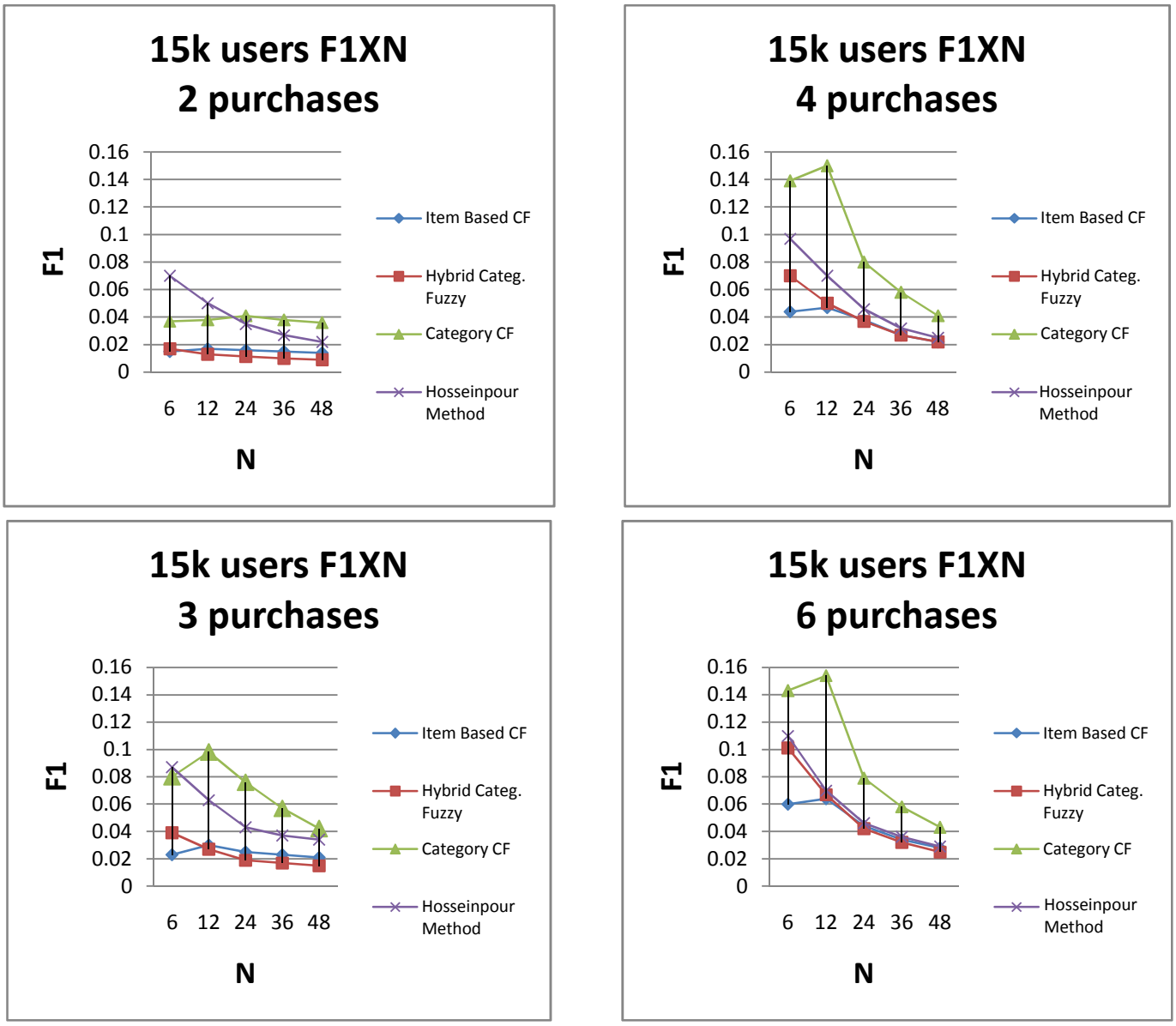

Figure 7. Experiments for several values of top-N by changing the number of a user's purchases (training for 15,000 users).

[7] S. J. Hoch and D. A. Schkade, "A psychological approach to decision support systems," Manag. Science, vol. 42, pp. 51–64, January 1996.

[8] P. Resnick, N. Iacovou, M. Sushak, P. Bergstrom, and J. Riedl, "GroupLens: an open architecture for collaborative filtering of netnews," Computer Supported Collaborative Work Conf., pp. 175–186, 1994.

[9] U. Shardanand and P. Maes, "Social information filtering: algorithms for automating 'word of mouth'," Conference of Human Factors in Computing Systems, pp. 210–217, 1995.

[10] K. Goldberg, T. Roeder, D. Gupta, and C. Perkins, "Eigentaste: a constant time collaborative filtering algorithm," Information Retrieval, vol. 4, no. 2, pp. 133–151, 2001.

[11] B. N. Miller, J. A. Konstan, and J. Riedl, "PocketLens: toward a personal recommender system," ACM Trans. Information Systems, vol.22, pp. 437–476, 2004.

[12] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: item-to-item collaborative filtering," IEEE Internet Computing, vol. 7, pp. 76–80, 2003.

[13] T. Hofmann, "Latent semantic models for collaborative filtering," ACM Transactions on Information Systems, vol. 22, pp. 89–115, 2004.

[14] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl, "Analysis of recommendation algorithms for e-commerce," ACM E-Commerce Conference, pp. 158–167, 2000.

[15] M. Pazzani and D. Billsus, "Learning and revising user profiles: the identification of interesting web sites," Machine Learning, vol. 27, pp. 313–331, 1997.

[16] M. J. Hosseinpour, M. Mosalanezhad, I. Badrooh, and M. Mojarad, "An intelligent fuzzy-based recommendation system for consumer electronic products," 3rd Int. Conference on E-commerce with focus on developing countries, pp. 22-23, 2008.

[17] D. H. Kraft, J. Chen, M. J. Martin-Bautista, and M. A. Vila, "Textual information retrieval with user profiles using fuzzy clustering and inferencing," in Szczepaniak, Segovia, Kacprzyk, and Zadeh (eds.), Intelligent Exploration of the Web, Physica-Verlag, 2002.

[18] O. Nasraoui and C. Petenes, "An intelligent web recommendation engine based on fuzzy approximate reasoning," FUZZ-IEEE Conference, vol. 2, pp. 1116-1121, 2003.

[19] Z. Zhang and O. Nasraoui, " Efficient hybrid web recommendations based on Markov clickstream models and implicit search," IEEE/WIC/ACM Int. Conf. Web Intelligence, pp. 621-627, 2007.