2014 IEEE Congress on Evolutionary Computation (CEC)
July 6-11, 2014, Beijing, China

A Social-Evolutionary Approach to Compose a Similarity Function Used on Event Recommendation

Luiz Mário L. Pascoal
Institute of Informatics
Federal University of Goiás
Goiás, Brazil
Email: luizpascoal@inf.ufg.br

Celso G. Camilo Júnior
Institute of Informatics
Federal University of Goiás
Goiás, Brazil
Email: celso@inf.ufg.br

Edjalma Q. da Silva
Institute of Informatics
Federal University of Goiás
Goiás, Brazil
Email: edjalmasilva@inf.ufg.br

Thierson C. Rosa
Institute of Informatics
Federal University of Goiás
Goiás, Brazil
Email: thierson@inf.ufg.br

***Abstract*—With the development of web 2.0, social networks have achieved great space on the internet, with that many users provide information and interests about themselves. There are expert systems that use the user's interests to recommend different products, these systems are known as Recommender Systems . One of the main techniques of a Recommender Systems is the Collaborative Filtering (User based) which recommends products to users based on what other similar people liked in the past. However, the methods to determine similarity between users have presented some problems. Therefore, this work presents a proposal of using social variables in the composition of the similarity function applied to a user on the recommendation of events. To test the proposal, details of friends and events of two target-users of the social network Facebook have been extracted. The results were compared with different deterministic heuristics, the Euclidean Distance and a aleatory method. The proposed model showed promising results and great potential to expand to different contexts.**

I. Introduction

With the development of web 2.0, term popularized by O'Reilly Media in 2004, the Internet began to be used as a " platform" for creating spaces where people could express their opinions and preferences freely through blogs and other means [1]. Eventually, blogging evolved to Social Networks, where people have a way to project themselves on the Internet, addressing their tastes, preferences, routines and even feelings, generating a large volume of data. However, the main purpose of Social Networks is to promote social interaction among users, presenting a new paradigm on the Internet known as the Social Web.

The Social Web presents a set of social connections (relationships) among the users of the Web, where its components, websites or software, are designed to support and promote social interaction. These tools can be from different fields: long distance education, gaming and also the ecommerce [2]. With the rapid development of the Social Web concept, electronic commerce has changed its patterns. Nowadays, the largest seller's on the internet offer thousands of products and choosing one among them is becoming an increasingly difficult task for the user. With this, more and more people seek different kinds of tips before acquiring a product.

One way to filter this data volume is to receive recommendations from other people we trust (Word of Mouth) [3], or check the review from professionals, which is very common with films, books, radios and diferent kind of events. But, this strategy can only reach a limited number of people in a certain period of time. And besides products, users also seek services and events reviews.

An event can be defined as an "happening", whose main characteristic is to provide an extraordinary opportunity to gather people with a specific purpose which constitutes the main "theme" of the event and justifies its realization. Currently, there is a big diversity in types of events, whether it's entertainment, academics or professionals kind. This diversification causes a huge increase in the amount of events that a person can participate, thus to make the decision of which event to go becomes really complex, especially in entertainment events where the preferences of the event type can vary considerably on different people. It is well known that anyone prefers to attend to a event with some of his friends instead of going alone, this imply that friends have a strong influence on the decision on whether go or not to a specific event. Thus one can consider friendship a valid and important variable to suggest events.

Thus, the use of social variables, like age, gender or relationship status can help to improve a specialist system that assists the user in the decision-making process. Those systems are known as Recommender Systems (RS).

The RSs were created to filter the amount of data available, by selecting the content to be presented to the target-user (user that receives the recommendation). However, a major challenge for these systems is to properly select the content to a target-user, since the preferences are extremely diverse among the them [4].

Among the various approaches of RS for filtering information in order to generate recommendations for a user, a technique that has gained prominence by the ease of incorporation with other approaches is the Collaborative Filtering. Collaborative Filtering uses the preferences of the target-user, seeking to recommend products that other users with similar preferences of the target-user have expressed interest for in the past. An important component of Collaborative Filtering is the similarity function that determines how close the target-user is to his similars, and this is a factor that directly influences the generation of a good recommendation [5] [4] [6].

On event (as other products) recommendation there is an important question to be answered: "What's the best way to recommend?". As stated earlier, the friends represent a significant variable in recommending events. Discovering what makes a person to have a bond with someone else is a very complex issue, since it involves several different reasons, where it can be affinity in preferences, stories or memorable moments together, or even resemblance according to different social variables such as age, gender, relationship status, and many others. All these aspects tend to group people who have similar realities. So, such social variables could contribute in the event recommendation process, in a way that the combination of these variables expresses a similarity function between the target-user and his friends.

Therefore, this study aims to find the best arrangement of social variables to compose a customized similarity function between the user and his friends, in order to improve the event recommendation. The experiments conducted show that within the event context, for two different target-users, it was possible to find a similarity function composed by the weighting of low dynamic social variables, and also that the discovered function is capable of obtaining good results when applied in different periods of time, inside a very dynamic context.

The rest of the paper is structured as follows: Section II discusses the Related Work, Section III presents a brief theoretical survey of the main points in RS; Section IV describes the implemented model, Section V presents the experiments and results obtained; And finally we have the conclusions and future work.

II. Related Work

Some surveys on Recommender Systems are available in the literature, where topics are discussed from pre-processing the database to the accomplishment of the recommendation itself. As presented in the work of Adomavicius and Tuzhilin [5] where a detailed state of the art for Recommender Systems, comparing the main systems developed at the time of labor in relation to the technique used by each.

A widely used technique in Recommender Systems is Collaborative Filtering. This technique has proven useful in studies of Zhou (2010) [7], Tan and Ye (2009) [8], Bu and He (2010) [9], Jamil, Alhadi and Noah (2011) [10] and Mu e Jing (2010) [11].

Collaborative Filtering algorithms are based on tastes and actions we take on our daily bases, which also depend on the recommendations of others. Thus, methods that incorporate social network information may be relevant. This can be well observed in the work of Golbeck [12], which performs a different approach from conventional Collaborative Filtering algorithms and presents a website that uses the reliability of social networks on the web to establish how close a product is to the user's preference. There is also the work of Ryu et al. [13] that identifies the nearest neighbors of a user through the Pearson correlation and considers this set of neighbors as a social network. Finally, Liu and Lee [14] perform the combination of the traditional approach of Collaborative Filtering with the social model of data extracted from a social network called Cyworld where the results show improvements in the accuracy for predicting user's preferences and also that the use of social information lowers the computational cost for traditional algorithms.

The work Kayaalp, Ozyer and Ozyer [15] implements a system that recommends shows to the users of Facebook. All information about the shows on the internet are collected via scrappers, and recommendations are made based on users profile, their preferences, past evaluations, and the properties of the event to occur. To implement this system, the hybrid approach was used, where through Collaborative Filtering more users were reached, and the Content-Based technique was responsible for extracting the features of each show, such as gender and bands that will perform. The results obtained in this experiment indicate that the greater the number of events considered, the greater is the accuracy of the algorithm, gradually decreasing the proportion of events that should not be on the list of recommendations.

Every researcher knows how exhausting it is to go to a conference and have a bunch of presentations, with different topics, and indeed find researches that are relavantes for his line of work, which will add knowledge to it. Thinking about this, Pham et al. [16] implemented a context-aware recommender system for mobile devices, that recommends presentations, discussions and even other participants in the event of academic nature. The paper proposes a modified version of Collaborative Filtering, inferring the social context through user research activities. To evaluate the tool, the databases of publications DBLP and CiteSeerX were used to identify studies that interest the researcher. The simulation conference was held from the data collected from ICWL 2010 (International Conference on Web-based Learning 2010) and one study site in the EC-TEL 2011 (European Conference on Technology Enhanced Learning 2011). The results for both conferences proved very satisfactory.

The main differential of this work is the presentation of a model capable of determining a similarity function between different people based on their social variables. Because of the stability of those variables, the patterns of the established function can be applied at different points in the future, which brings great strength to a possible recommender method.

III. Recommender Systems based on Collaborative Filtering

Since the creation of the first recommendation system based on Collaborative Filtering named Tapestry [17], this area of research has received great attention from the academic community due to various practical applications that help users deal with large amounts of data or to ensure a customized recommendation. There is also a commercial interest, since the use of a recommendation system can provide a maximization on sales for a company. Examples of companies that have applications for recommendation are: Amazon, MovieLens, eBay and even Google [5].

The Recommender Systems help to increase the capacity and effectiveness of the suggestion process already performed in the social relationship between humans [4]. They represent the preferences allied to a number of additional data, in order to suggest products to user acquisition and analysis, as pointed in figure 1. In the study by Goldberg et al. [17] the concept of Collaborative Filtering was originally presented to designate a type of system that was able to filter the information through the cooperation of other human beings. The idea behind the collaboration is to somehow grouping people who will receive recommendations from people who have previously interacted with the system since they exhibit a significant degree of similarity [5].

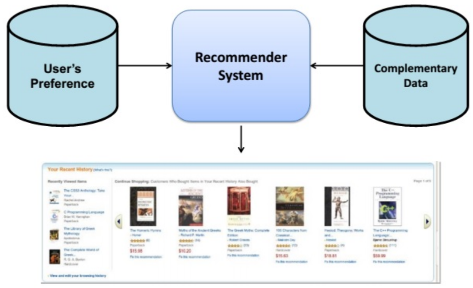

Fig. 1. Simple Flow of a Recommendation System

To accomplish such grouping, it is necessary to have knowledge about the user. This knowledge can be obtained explicitly, where the user openly tells the system what his interests are, or may also be obtained implicitly, through analyzes of the behavior of users while interecting with the system (most clicked products, time on a given page, etc). So, who presents most common interests or behavior will be properly grouped.

This grouping of people is done based on a degree of similarity between them. The degree of the similarity is determined by a similarity function, through mathematical calculations that measure the distance between users about n dimensions. In the literature we have several consecrated functions that are also widely used in the area of Recommender Systems. Three classic examples are: the Cosine Function, widely used in Content Based Filtering, where weights of terms are treated as vectors and the similarity is determined by the cosine of the angle between them; Euclidean distance, shown in equation 1 for the $n - dimensional$ Euclidean model, where p and q represent dimensions (variables).

$$d = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + ... + (p_n - q_n)^2} \quad (1)$$

Another method widely used is the Pearson correlation, shown in equation 2. It is used to find the $Top - N$ nearest neighbors with similar interests. Thus, we have a and b represent two people, j an ordinary product, $v_{a,j}$ is the evaluation of the user a on item j. The total average ratings given by $\overline{v}_a$, performed by the user a in the system can be obtained through the equation 3 [5] [14].

$$\omega(a,b) = \frac{\sum_j (v_{a,j} - \overline{v}_a)(v_{b,j} - \overline{v}_b)}{\sqrt{\sum_j (v_{a,j} - \overline{v}_a)^2 \sum_j (v_{b,j} - \overline{v}_b)^2}} \quad (2)$$

$$\overline{v}_a = \frac{1}{|I_a|} \sum_{j \in I_a} v_{a,j} \quad (3)$$

Among the various existing techniques for Recommender Systems, we focus on Collaborative Filtering given the fact that is easy to incorporate social information to the model. As previously stated, Recommender Systems based on Collaborative Filtering generate recommendations to users about products that other users with similar tastes and preferences have liked in the past. The most common approach is to find the nearest neighbors by calculating the Pearson correlation coefficients (Eq. 2). Upon finding the $Top - N$ nearest neighbors, it is possible to make predictions about the user's opinion in relation to a particular product [18]. In equation 4, the predicted evaluation of the user a for the product j is given by $P_{a,j}$ and can be calculated by using the weighted average of all evaluation for product j, and the average of all ratings made by user a on other products, as $W_{(e,a)}$. $Raters$ are the set of users who evaluated product j [14].

$$P_{a,j} = \overline{v}_a + \frac{\sum_{e \in Raters} (v_{e,j} - \overline{v}_e) W_{(e,a)}}{\sum_{e \in Raters} |W_{(e,a)}|} \quad (4)$$

The evaluation of the prediction for user a about the product j is used to determine the possibility of recommending the item j to user a. If the prediction presents a low value, it is less likely that the product is going to be recommended to the user. However, nothing prevents the user to rate product j in the future and the result of the evaluation is completely different from the predicted evaluation, therefore it is possible to calculate the accuracy of the prediction method.

Systems that work with the collaborative approach can recommend any product, even if it is completely different from those products previously recommended. However, this approach has some limitations, such as: **New User:** To perform a precise recommendation, the system first needs to learn the user's preferences through the classification or evaluation that he does. But when the user first starts using the system, these informations are not available, making it impossible to generate a personalized recommendation for that profile. The proposal of this work intends to generate a method for dealing with this problem; **New Product:** A product that has just been registered in the system has no reviews, so it may not be recommended by the system. Or it can take a long time for this new product to recieve an equivalent reputation to the most popular products, so it has a low priority in recommendation; **Sparse Ratings:** In any Recommender System, the number of reviews is commonly very small compared to the number of evaluations needed to make a good prediction. We also have the case of users that don't exhibit the same tastes as most of other users. Since they are the minority, the system will collect few evaluations which will culminate in weak recommendations [5] [19].

IV. Proposal

A. Model Overview

There are many variables that make a person interested in any particular event. Variables such as personal preferences, the kind of the event, schedule availability, the people coming to the event and others. These are key points in the user's decision to participate or not in an event. The bond of friendship is a good indicator of interest, since this connection employs different levels of intimacy and reliability between two or more people. Based on this principle, the proposed method uses the bond of friendship between a user and his friends to determine a customized similarity function that checks which set of social variables represent best the user's preference for events.

With the popularity of social networks, especially Facebook, we have a digital tool to manage and represent these social bonds of friendship, making it easier to collect a lot of information. However, a user has a large amount of friends on his list of Facebook friends and not all of them influences the user's opinion. Therefore, it was necessary to filter his friends list to determine who were the most significant friends in order to recommend events.

There are different approaches to filter the list of friends. In this paper, the friends on the list were ordered according to the frequency in different events, in order to determine Hubs from the user's friend list, that is, who are the user's friends who most frequent different events. However, the friend's list filtering could also be done manually, where the user explicitly states who are his best friends on his friends list. The filtered list was called Control List (CL). Thus, this project was divided into two stages:

- **Step 1:** Implementation of a Web System that through the Facebook API [20], retrieves the data from the target-user and his friends, filtering the list of friends to discover who has higher frequency in different events. The friends who are most frequent in different events will compose the CL.
- **Step 2:** Construction of a Genetic Algorithm that attempts to reproduce the CL through the linear combination of social variables available for each friend on Facebook. Functions extracted from individuals of the Genetic Algorithm are similarity functions.

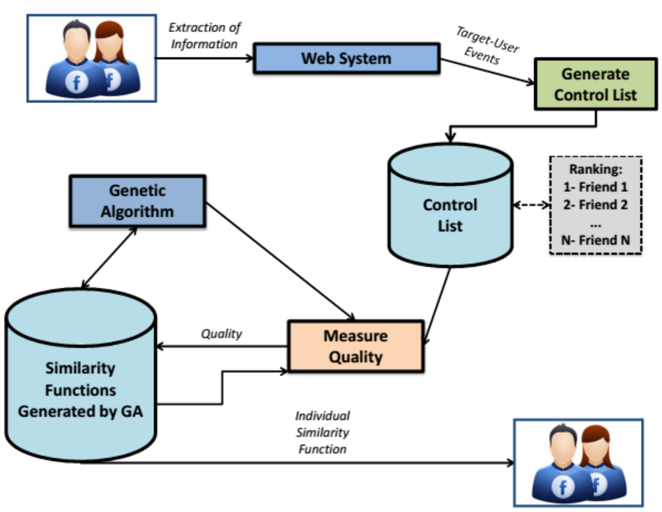

Fig. 2. General Architecture of the Proposed Method

In figure 2 it is presented a diagram that shows the flow of proposed method. First, the user authenticated by Facebook, allows access to his personal data. The Web System extracts the information necessary to compose the CL. In this paper, friends who have a higher frequency in different events are stored in the CL, and will be ordered according to their amount of participation in different events. All people present in the CL will have their social information extracted to feed the Genetic Algorithm.

After this step, the genetic algorithm receives the CL generated by the Web System, and ponders the social variables so that it can attempt to reproduce the list. The best combination of variables will result in a specific similarity function for the target-user and his friends.

It is noteworthy that the CL is created with a similarity function based on the frequency of friends in different events. The Genetic Algorithm intends to find a similarity function that approximates the function that generated the CL. However, the Genetic Algorithm uses only social variables of the target-user friends. If the Genetic Algorithm can discover a function that approximates to the CL, we have a method that generates similarity functions, composed of social variables, which are more stable and may be used in future times.

B. GA Modeling

Genetic Algorithms (GA) are search and optimization methods inspired by the mechanisms of evolution in the populations of living beings. They were first introduced by John Holland in 1975 and popularized by one of his students, David Goldberg in 1989. These algorithms follow the principle of natural selection, survival of the fittest, declared in 1859 by British naturalist and physiologist Charles Darwin in his book, The Origin of Species. According to Charles Darwin, " The better an individual to adapt to its environment, the greater is its chance to survive and to generate offspring" [21].

Basically, what a GA does is create a population of possible solutions to the problem in question and then submit it to the evolutionary process, as it follows [22]:

- **Evaluation:** Evaluates the fitness of the solutions (individuals in the population);
- **Selection:** Individuals are selected for reproduction. The probability of a given solution i be selected is proportional to its fitness;
- **Crossover:** Characteristics of the chosen solutions are recombined, generating new individuals;
- **Mutation:** Characteristics of individuals arising from the reproducing process are changed, thereby adding variety to the population;
- **Update:** The new individuals are inserted into the population;
- **Finishing:** Checks whether the termination conditions were achieved. If yes, ends the execution. If not, returns to Evaluation step.

In this paper problem, The GA will receive L lists and should learn a similarity function that is able to reproduce them through the weighting of social attributes, as characterized in equation 5, which will hold the sum of the difference between the positions calculated for the first k friends presented in the CL, given as $A_iP_{Real_n}$, and the position that these same k friends occupy on the list generated by the GA, given as $A_iP_{GA_j}$. So, we obtain the accuracy of the GA individual by comparing the original position of the friends on the CL and the position that these same friends occupy in the GA ordering.

$$MIN(f(x)) = \sum_{1}^{L} \left(\frac{\sum_{1}^{k} \left| A_iP_{Real_n} - A_iP_{GA_j} \right|}{L} \right) \quad (5)$$

The $List_{GA}$ consists of the ordering of the target-user's friends by the score given by the equation 6, for each friend i in the CL. The calculation in equation 6 is performed to quantify the similarity coefficient of friend A_i, expressed by $Evaluation_{A_i}$, where: each W_n is the value of the gene n of the chromosome generated by GA, and X_n is the normalized numerical value for each social variable.

$$Evaluation_{A_i} = \sum_{1}^{n} W_n X_n \tag{6}$$

In equation 6, X_n is the numerical value for each social variable of the A_i friend in CL. In this paper, the social variables used are: Age, Gender, Education, and Relationship Status. However, not all social variables selected are numeric values, as in the case of Gender, Education and Relationship, so it has been adopted a set of numerical values for these variables (expressed in parenthesis) that maps the non-numeric variables into numerical sets. Once all variables are mapped, it is necessary to normalize all variables so they can be on the same scale of values. The social variables were categorized according to the data presented on Facebook, and may assume values as listed below:

- **Age:** Integer Value;
- **Gender:** Male (1) or Female (2);
- **Education:** High School (1), College (2) or Graduate School (3);
- **Relationship:** Single (1), In a Relationship (2), Engagement (3) or Married (4);

Still in equation 6, we have W_n used to weight the social variables, where each gene n is the weight of each social variable. So, the chromosome is defined with four genes, as shown in figure 3 with the chromosomal representation, but the model also works with a greater number of variables. Each gene in the generated chromosome can assume values from -1 until 1.

W_{age}	W_{gender}	$W_{education}$	$W_{relationship}$

Fig. 3. Chromosome Representation

V. EXPERIMENTS AND RESULTS

This section discusses the experiments and results obtained with them. The experiments were performed with data from two distinct profiles, known here as 'User 1' and 'User 2'.

The system uses the database available on the Facebook API [20] as a data source for events and profile information of the user's friends. The data were collected on November 21st, and all event data is for the year 2013.

The extracted data are used to construct different Control Lists (CL). It is worth noting that each generated CL is different because each CL displays data for different seasonal periods of 2013. I.e, the CL of a particular month is composed by the ranking of friends who most attended to different events that month. An example of a CL is shown in the table I, which has the November's CL for 'User 1'.

TABLE I. USER 1 - NOVEMBER CONTROL LIST

Name	Age	Gender	Education	Relationship
Friend 1	24	Male	College	In a Relationship
Friend 2	24	Male	College	In a Relationship
Friend 3	24	Male	College	In a Relationship
Friend 4	22	Male	College	Single
Friend 5	21	Male	College	In a Relationship
Friend 6	22	Female	College	In a Relationship
Friend 7	24	Female	College	In a Relationship
Friend 8	24	Male	College	Single
Friend 9	21	Female	Graduate	In a Relationship
Friend 10	24	Male	Graduate	In a Relationship

A. Experiment One - GA Calibration

The structure of a GA allows it to use various operators in phases, where their combination have a big impact on the result and to the computational cost to reach the best result. Because of that, this experiment was designed to select which selection operator, Roullete (R) or Tournament (T) [23], and which mutation rate is the best setting to obtain the best performance for both profiles.

For this experiment, the GA was executed for two different CLs containing acumulative data, where the first CL is composed by friends who most attended to different events that occurred in January until March, and the second CL is composed by friends who most attended events from January until June, varying the selector operators (Roullete and Tournament) and also the mutation rate.

The GA is a stochastic method, so it was executed 30 times and it was calculated the average error for the results to 'User 1' and 'User 2' after 30 executions. The results are presented in table II:

TABLE II. MEAN ERROR FOR EACH USER

	Mutation Rate - 2%		Mutation Rate - 5%		Mutation Rate - 10%	
User	**R**	**T**	**R**	**T**	**R**	**T**
User 1	7,93	7,87	7,87	7,97	7,87	7,87
User 2	33,80	33,93	33,27	33,10	33,53	33,60

Based on table II, we define the best combination for the operators, considering both profiles, which was the Roullete selection operator and mutation rate in 5%. Therefore, we define the GA's configuration as described below:

- **Population:** 100 individuals;
- **Number of Generations:** 1000 generations;
- **Crossover Probability:** 80%;
- **Mutation Rate:** 5%;
- **Elitism:** Worst, the worst individuals are substituted [24];
- **Selection Operator:** Roullete [25];
- **Crossover Operator:** Intermediate - operator used on float problems [25];

B. Experiment Two - Seasonal GA based on Training and Testing

The proposed model for Experiment Two divides its operation into two phases: Training and Testing. In the training phase, the GA is executed with the purpose of learning a similarity function between the target-user and his friends by weighting their social variables. At this phase, two CLs are used, where the first contains event data from the period of January until March, and the second from January until June. The second list contains the cumulative data from the first list, this is performed so that the GA could learn a more consistent and robust function over time. The testing phase consists of applying the similarity function learned by the GA in the training phase in each of the following months from July until November.

A major difference between the two users is the amount of friends presented in each CL, whether in training or testing. It is worth mentioning again that each month features a different CL because it is made only by friends who are more frequent in different events in that particular period, as shown in the table III.

TABLE III. NUMBER OF FRIENDS

Months	User 1	User 2
Training		
January - March	16	25
January - June	18	30
Testing		
July	15	27
August	17	21
September	17	23
October	11	18
November	10	18

Once the GA parameters are defined, we can start the execution of the proposed model. This experiment had the purpose to verify the performance of a learned function based on social variables in the following months after the learning phase. For this experiment, we set $k = 5$ on the computation of equation 5 which represents GA's fitness.

To prove the functionality and efficiency of the proposed model, we use the GA's learned function during the training period into the testing phase. Also, to evaluate the performance of the learned function, all the results obtained will be compared to different heuristics listed below, designed with the objective of minimizing the computational cost and explore a new possible solution that uses a smaller amount of social variables. The results are also compared to the Euclidean distance, which is a function widely used in the field of recommendation systems, and also compared to an adaptation for this function, known as Weighted Euclidean Distance.

- **Used Heuristics:**
  - Ranking by Age using Education to break the tie (AdE);
  - Ranking by Age using Relationship to break the tie (AdR);
  - Ranking by Education using Age to break the tie (EdA);
  - Ranking by Education using Relationship to break the tie (EdR);
  - Ranking by Relationship using Age to break the tie (RdA);
  - Ranking by Relationship using Education to break the tie (RdE);
  - Ranking by Shuffling the CL (Random);
  - Ranking based on Euclidean Distance - which measures the distance between the target-user and each friend from CL regarding their social variables;
  - Ranking based on Weighted Euclidean Distance - which measures the distance between the target-user and each friend from CL regarding their social variables and uses the similarity function learned by the GA to weight the importance of each social variable;

The metric used to evaluate the comparison methods is described by the equation 5, replacing $A_i P_{AG_j}$ by the position that the friend A_i on CL occupies in the ranking generated by each heuristic.

The tables V and VII show the calculated fitness for each heuristic used, so it is possible to make a comparison between the approaches and verify the effectiveness of each one. We called Test-GA the proposed model based on Training and Testing. It was also implemented a new version of a GA, named New-GA, where the goal is to observe the behavior if the CL was trained every month, eliminating the division of Training and Testing proposed in the previous model, which means that New-GA learns a similarity function between the target-user and each friends on CL for the Testing period.

Two other methods were also implemented for comparison of results, the Euclidean Distance and the Weighted Euclidean distance. Both methods use the social variables of the target-user and his friends, but the $Evaluation_{A_i}$ for every friend is calculated from the Euclidean Distance, shown in equation 1, between the target-user and each friend presented on CL regarding their respective social variables. The Weighted Euclidean Distance method uses the similarity function learned by Test-GA as weight of each social variable in the computation of Euclidean Distance.

All stochastic methods were executed several times due to the randomness in the creation of the solution. So, the result was calculated from the average fitness value in relation to the total of executions. In the Training phase, the Test-GA was executed 30 times for each target-user, in the end it was extracted the best similarity function for the users 'User 1' and 'User 2', as shown in Tables IV and VI respectively. In the Testing phase, the New-GA was executed 30 times and the random method 1.000 times, the results presented in tables V and VII are an average of the executions. The Test-GA, the Used Heuristics, Euclidean Distance and the Weighted Euclidean Distance were executed only once given its deterministic features. Also, all the results shown in tables V and VII were properly normalized according to the maximum possible error given the amount of friends in each CL for every month of the Testing period.

TABLE IV. USER 1: TRAINING PHASE - LEARNED FUNCTION

0.4598	-0.4196	-0.3046	-0.0154	**Mean Error:** 7,5

TABLE V. USER 1: APPLICATION OF THE LEARNED FUNCTION - TESTING PHASE

	Results for User 1				
	July	**August**	**September**	**October**	**November**
Test-GA	0.28	0.27	0.27	0.67	0.48
New-GA	0.152	0.18	0.18	0.27	0.08
AdE	0.52	0.50	0.27	0.7	0.44
AdR	0.44	0.42	0.28	0.7	0.48
EdA	0.52	0.55	0.35	0.87	0.64
EdR	0.44	0.53	0.37	0.97	0.60
RdA	0.32	0.33	0.32	0.67	0.48
RdE	0.32	0.40	0.33	0.77	0.60
Random	0.56	0.54	0.54	0.76	0.66
Euclidean	0.38	0.43	0.40	0.73	0.52
Weighted Euclidean	0.70	0.50	0.58	0.90	0.80

The table V presents the results for 'User 1' with all the previously mentioned heuristics. The table shows that the Test-GA, has achieved good results throughout the Testing period compared with the other established heuristics, being second only to New-GA that learns a similarity function for each month, proving that the Test-GA learned function, shown on table IV, can be used efficiently in future moments. In the month of September, the AdE heuristic was able to achieve the same result as Test-GA, which implies that age is a strong variable to sort k friends on CL, as stated by the similarity function learned on table IV. It is also worth noting the poor performance of both Euclidean distances methods, which shows that the target-user is very different from the main hubbies of his friends list, in relation of their social variables.

TABLE VI. USER 2: TRAINING PHASE - LEARNED FUNCTION

0.4399	0.2745	-0.0137	0.7563	**Mean Error:** 31,5

TABLE VII. USER 2: APPLICATION OF THE LEARNED FUNCTION - TESTING PHASE

	Results for User 2				
	July	**August**	**September**	**October**	**November**
Test-GA	0.53	0.33	0.29	0.43	0.32
New-GA	0.24	0.32	0.25	0.32	0.12
AdE	0.56	0.37	0.59	0.38	0.49
AdR	0.60	0.4	0.59	0.46	0.49
EdA	0.52	0.41	0.58	0.34	0.6
EdR	0.48	0.42	0.51	0.43	0.49
RdA	0.62	0.41	0.49	0.52	0.4
RdE	0.6	0.41	0.48	0.52	0.43
Random	0.52	0.52	0.52	0.53	0.53
Euclidean	0.54	0.60	0.49	0.45	0.62
Weighted Euclidean	0.51	0.59	0.54	0.43	0.71

The table VII shows the results for the 'User 2'. It is easy to observe that the 'User 2"s learned function (table VI) obtained a reasonable result when applied directly on each CL, as the results shown for Test-GA. However, when we apply the learned function on the Euclidean Distance, given as Weighted Euclidean, we obtain a better result, which proves the efficiency of the learned similarity function. This happens because the target-user social variables are similar to his friends in each CL.

Based on the results obtained for users 'User 1' and 'User 2', we can conclude that the combination of all social variables presents better results than using fewer variables. Equally, improving the original Euclidean Distance equation adding the weights of the function learned by Test-GA to calculate the distance between the target-user and his friends has proven to be successful. This shows that all social variables do not have the same influence on the good result, which means that some variables should be favored in place of others. Finally, we observed the excellent performance of New-GA in obtaining the better result for both profiles, but its use involves a high computational cost required for a GA standard operations.

VI. CONCLUSIONS AND FUTURE WORK

This study had as main objective to find the best arrangement of social variables, in the form of a customized similarity function between the target-user and his friends in order to improve the event recommendation. The experiments conducted show that with few social variables it is possible not only to define a similarity function between the target-user and his friends, but it is also possible to use it in future moments, creating the possibility of using some low dynamic variables, which have proved very effective to predict and evaluate objects in a very dynamic environment such as events, showing the great potential of the proposed model.

As future proposals, it is intended primarily to collect data from many different users and carry out further tests to prove the robustness of the model on events. We also intend to improve the model by including behavioral variables within social networks. Then expand and test the improved model created into new fields, such as movies, music, books, among others, and group the results obtained in these different fields, to determine a more accurate and robust similarity function in relation to different areas.

ACKNOWLEDGMENT

The authors would like to thank CAPES for the support with a graduate student stipend, and also thank FAPEG - Fundação de Apoio à Pesquisa do Estado de Goiás, that partially supported this work, under grant Chamada 009/2010.

REFERENCES

[1] T. O'reilly, "What is web 2.0," 2005.

[2] D. Appelquist, D. Brickley, M. Carvahlo, R. Iannella, A. Passant, C. Perey, and H. Story, "A standards-based, open and privacy-aware social web," *W3C Incubator Group Report*, vol. 6, 2010.

[3] U. Shardanand and P. Maes, "Social information filtering: algorithms for automating "word of mouth"," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM Press/Addison-Wesley Publishing Co., 1995, pp. 210–217.

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

[5] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 17, no. 6, pp. 734–749, 2005.

[6] K. Verbert, N. Manouselis, X. Ochoa, M. Wolpers, H. Drachsler, I. Bosnic, and E. Duval, "Context-aware recommender systems for learning: a survey and future challenges," *IEEE Transactions on Learning Technologies*, 2011.

[7] M. Zhou, "Book recommendation based on web social network," in *Artificial Intelligence and Education (ICAIE), 2010 International Conference on*. IEEE, 2010, pp. 136–139.

[8] H. Tan and H. Ye, "A collaborative filtering recommendation algorithm based on item classification," in *Circuits, Communications and Systems, 2009. PACCS'09. Pacific-Asia Conference on*. IEEE, 2009, pp. 694–697.

[9] X. Bu and X. He, "An optimized trust factor based collaborative filtering recommendation algorithm in e-commerce," in *Information Science and Management Engineering (ISME), 2010 International Conference of*, vol. 1. IEEE, 2010, pp. 468–471.

[10] N. Jamil, A. Alhadi, and S. Noah, "A collaborative names recommendation in the twitter environment based on location," in *Semantic Technology and Information Retrieval (STAIR), 2011 International Conference on*. IEEE, 2011, pp. 119–124.

[11] Z. Mu, L. Jing, C. Shan, and F. Lei, "Design of the tourism-information-service-oriented collaborative filtering recommendation algorithm," in *Computer Application and System Modeling (ICCASM), 2010 International Conference on*, vol. 13. IEEE, 2010, pp. V13–361.

[12] J. Golbeck, *Generating predictive movie recommendations from trust in social networks*. Springer, 2006.

[13] Y. Ryu, H. Kim, Y. Cho, and J. Kim, "Peer-oriented content recommendation in a social network," in *16th workshop on informaton techonologies and systems (WITS 2006)*, 2006.

[14] F. Liu and H. J. Lee, "Use of social network information to enhance collaborative filtering performance," *Expert systems with applications*, vol. 37, no. 7, pp. 4772–4778, 2010.

[15] M. Kayaalp, T. Ozyer, and S. Ozyer, "A collaborative and content based event recommendation system integrated with data collection scrapers and services at a social networking site," in *Social Network Analysis and Mining, 2009. ASONAM'09. International Conference on Advances in*. IEEE, 2009, pp. 113–118.

[16] M. C. Pham, D. Kovachev, Y. Cao, G. M. Mbogos, and R. Klamma, "Enhancing academic event participation with context-aware and social recommendations," in *Advances in Social Networks Analysis and Mining (ASONAM), 2012 IEEE/ACM International Conference on*. IEEE, 2012, pp. 464–471.

[17] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," *Communications of the ACM*, vol. 35, no. 12, pp. 61–70, 1992.

[18] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems (TOIS)*, vol. 22, no. 1, pp. 5–53, 2004.

[19] K.-W. Cheung, K.-C. Tsui, and J. Liu, "Extended latent class models for collaborative recommendation," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 34, no. 1, pp. 143–148, 2004.

[20] W. Graham, *Facebook API developers guide*. Infobase Publishing, 2008.

[21] D. E. Goldberg and J. H. Holland, "Genetic algorithms and machine learning," *Machine learning*, vol. 3, no. 2, pp. 95–99, 1988.

[22] M. Mitchell, *An introduction to genetic algorithms*. MIT press, 1998.

[23] D. E. Goldberg and K. Deb, "A comparative analysis of selection schemes used in genetic algorithms," *Urbana*, vol. 51, pp. 61 801–2996, 1991.

[24] D. Thierens and D. Goldberg, "Elitist recombination: An integrated selection recombination ga," in *Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on*. IEEE, 1994, pp. 508–512.

[25] F. Herrera, M. Lozano, and J. L. Verdegay, "Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis," *Artificial intelligence review*, vol. 12, no. 4, pp. 265–319, 1998.