CAPTURING USER'S PREFERENCES USING A GENETIC ALGORITHM

Determining Essential and Dispensable Item Attributes

S. Valero, E. Argente and V. Botti
DSIC, Universidad Politécnica de Valencia, Camino de Vera s/n, Valencia, Spain

Keywords: Soft computing, Genetic algorithms, Recommender systems, Profile acquisition, Capturing preferences.

Abstract: Determining the most desired product attributes would be crucial for companies that want to offer their clients those products which best fit their preferences. In this work, a genetic approach is employed for establishing the appropriate attribute weights of movies, determining which movie attributes are essential or dispensable for users in their selection process. The obtained weights are employed to predict user's ratings for a test set of movies, proving that the obtained parameters really describe their preferences.

1 INTRODUCTION

Setting the most desired product attributes would be very interesting for any companies which want to offer its clients such items that best fit these desired features. In this way, it would be interesting to know which item attributes are essential or superfluous for users in their selection process when buying products or paying for services. This problem could be seen as a combinatorial problem, in which attribute based profiles of items are used as start point to set the appropriate attribute weights that make these products attractive for users. Thus, a soft computing approach based on genetic algorithms could be used to set the optimum combination of the item attribute weights that best fit the user's preferences. Some Soft Computing approaches have previously been applied in user's preference learning, such as (Guan et al., 2002; Shibata et al., 2002). These works capture users' preferences but do not consider some starting problems caused by missing information of new users. In this way, a genetic algorithm (GA) capable of establishing the appropriate weights for item attributes that define the users' preferences, but needing few starting data is presented in this work. Concretely, this GA determines the suitable combination of attributes that makes an specific item attractive for users. These combinations are later used to predict users' rating movies, making recommendations about unknown movies. Moreover, our approach needs few users' starting ratings (only 3 or 5) to obtain good recommendations (right rates closer to 75%). For this purpose, the Movilens database will be used, which contains ratings of movies made by MovieLens web site[1] users in 2000 (Herlocker et al., 1999).

2 GENETIC APPROACH

The proposed approach for capturing users' favorite movie attributes is based on a real-coded GA (Goldberg, 1991). The developed codification problem allows studying diverse kinds of problems and establishing rules at different levels that the obtained chromosomes must satisfy (Valero et al., 2009). In this case, each possible solution of the problem (chromosome) contains a descriptive movie attribute weight (release data) and genre attribute weights (19 possibilities: action, adventure, animation, children, comedy, crime, etc.).

Figure 1 shows the steps followed by the developed GA. First, the GA obtains a starting generation, selecting the most diverse one from several random generations. Then, the GA proposes new solutions employing crossover and mutation operators. The mutation operator modifies some genes in a haphazard way, jumping randomly anywhere within the allowed gene domain. Regarding the crossover operator, it has been adapted from the ones proposed in (Ortiz et al., 2001), which is based on confidence intervals. The new operator takes into account the rules defined in the developed codification. This GA ap-

[1] http://movielens.umn.edu

Valero S., Argente E. and Botti V. (2010).
CAPTURING USER'S PREFERENCES USING A GENETIC ALGORITHM - Determining Essential and Dispensable Item Attributes.
In *Proceedings of the 2nd International Conference on Agents and Artificial Intelligence - Artificial Intelligence*, pages 599-602
DOI: 10.5220/0002727305990602

proach has been successfully employed in combinatorial catalysis for high-throughput experimental design (Valero et al., 2004b; Valero et al., 2004a; Valero et al., 2009).

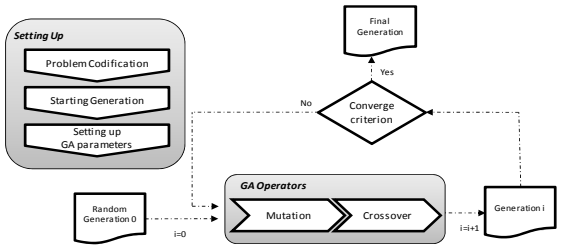

Figure 1: Steps of the developed genetic approach.

As mentioned above, the GA has been used to establish the appropriate attribute weights for movies that define the preferences of a set of one or more users. These obtained weights were employed to predict users' ratings for a test set of movies, in order to prove that the obtained parameters really described their preferences. Absolute Mean Error (MAE) and Mean Square Error (MSE) were used as metrics ((Sarwar et al., 1998; Shardnand and Maes, 1995; Herlocker et al., 1999)). The objective function optimized by the proposed GA was:

$$MAE = \frac{\sum_{i=1}^{M} |P_i - R_i|}{M} \quad (1)$$

where M indicates the number of users' movie ratings; P_i represents the estimated user rating for the movie i; R_i is the user's rating for the movie i. Function 2 describes the way in which P_i is obtained:

$$P_i = \sum_{j=1}^{N} w_j q_j \quad (2)$$

where w_i is the adjusted weight for the attribute j, q_i is the value of the attribute j for the movie to be rated and $\sum_{j=1}^{N} w_j = 1$. Thus, the fitness of each possible solution or chromosome was $\frac{1}{MAE}$. Finally, the developed GA proposed those attribute weights that best fit users' interests.

3 RESULTS

A user contained in the *MovieLens* database was selected taking into account the amount of available information about him. This user was employed to set up the GA parameters. Then, the GA was used for capturing the preferences of a cluster of users. These weights were later used to estimate both cluster ratings and ratings made on some other movies by similar users.

3.1 GA Setting Up Process

One user (identifier 276) with a lot of ratings (518) was selected, hence well informed data was used for the GA setting up process. The available information of user 276 was divided into training (90%) and testing (10%) data sets. Training data was used by the GA for fitness evaluation (function 1), whereas the testing data set was employed to predict the user's ratings. The testing data set contains information about recently rated movies. First of all, parameters concerning population size and crossover operator were set: α value, parents ratio and population size. A battery of tests was performed using all possible combinations of the values: population size$= \{100, 200, 300, 400, 500\}$; α value $= \{0.6, 0.7, 0.8, 0.9\}$; parents ratio$= \{10\%, 20\%, 30\%\}$. Mutation probability was set to 0; thus, the mutation operator did not act in these tests. The proposed GA reached 25 optimization cycles for each parameter combination (convergence criterion). At the end of each test, the best chromosome (attribute weights with best fitness) was selected in order to predict the user's ratings on movies in the testing data set. Also, the number of right movie recommendations was computed. A right recommedation was computed when the predicted rating for a movie indicated that the user would really like that movie and the stored rating at the database was also favorable ($rating \geq 0.5$ in both cases). Similarly, a right recommendation was also computed when the predicted rating for a movie shown a user dislike for the movie and its stored rating was also desfavorable ($rating < 0.5$ in both cases). Besides right recommendations, MAE and MSE were also considered to select the most suitable values for the studied parameters.

Table 1: Best results obtained during the optimization of the crossover and population parameters.

Parameters	MAE	MSE	Right Recom.
500-0.7-10	**0.2885**	**0.1250**	**69.23%**
500-0.6-10	0.3029	0.1358	63.46%
100-0.6-10	0.3413	0.1550	50.00%
400-0.7-30	0.3413	0.1550	50.00%
400-0.9-30	0.3413	0.1550	50.00%

Best results obtained during the reported study are shown in Table 1. The selected combination of parameters was: population size=500; α value=0.7 and parents ratio=10. The GA (using them) proposed a set of attribute weights which reached a 69.23% rate of right recommendations and the least MAE and MSE at the test phase.

Secondly, mutation parameter values (mutation probability and number of genes to be mutated) were

Table 2: Best results obtained during the optimization of the mutation parameters.

Parameters	MAE	MSE	Right Recom.
15% - 2	**0,2644**	**0,0925**	**78,85%**
15% - 3	0,2645	0,0974	76,92%
15% - 1	0,2836	0,1166	69,23%
5% - 3	0,2836	0,1166	69,23%
10% - 3	0,2788	0,1179	69,23%

set following the above-mentioned process. The values studied for these parameters were: mutation probability= $\{5\%, 10\%, 15\%\}$; number of genes to be mutated= $\{1, 2, 3\}$. Crossover and population parameter values were those previously selected ones. Obtained results are shown in Table 2.

The GA approach was finally parametrized as follows: population size = 500; α value = 0.7; parents ratio = 10; mutation probability=15% and 2 genes to be mutated. Using these parameters, the GA proposed a set of attribute weights which performed a rate of 78,85% of right recommendations. These attribute weigths indicated that the essential movie attributes for user 276 were the release date (19.32%) and action (23.51%), comedy (19.93%) and drama (20.73%) movie genres. In the same way, the dispensable movie attributes (0%) were animation, musical, mystery, romance, science fiction, war western and unknoun movie genres.

3.2 Employing Clusters Preferences

In this section the proposed GA is used to capture the preferences of system users which are similar to a reference user, employing few ratings of this reference user (i.e. 3 movie ratings). This approach can be used by recommender systems to offer acceptable suggestions when few starting data for a user is available.

Clusters of similar users were obtained using *Pearson* correlation as a proximity measure and center-based as a neighborhood scheme (Sarwar et al., 2000). All the data about clusters movie rating was divided into training and testing data sets. Then, the GA was employed for getting the appropriate movie attribute weights which define the preferences of these clusters (using training data for fitness evaluation). These optimized weights were later used to estimate reference users' ratings on the 20 movies recently rated by them. In this study, several experiments were carried out using two different reference users: user 276 (518 ratings), previously employed, and user 710 (86 ratings), selected in a haphazard way.

For each reference user, his first rated movies where considered to obtain his cluster of similar users. Each cluster is composed of users that have seen the same films and have evaluated them in a similar way with regard to the reference user. More specifically, clusters were formed considering three (3M) or five (5M) reference users' movie ratings and neighborhoods of 10, 15 or 20 users (k). Six clusters were computed for each user, but since for user 710 all clusters obtained with 3M were equal to those obtained with 5M, then less tests were performed for this user.

The optimized movie attribute weights proposed by the GA for each computed cluster were later used to estimate the cluster ratings on the movies in the testing data set. The results obtained in these estimations are shown in Table 3.

These appropriate movie attribute weights that define the cluster preferences were also used to predict the ratings on the 20 movies recently ranked by similar users. This study simulated the way in which real users act in Recommender Systems. In such environments, users enter few starting information (3M or 5M in this case) but expect good recommendations. Table 4 describes the achieved results. It should be pointed out the excellent results which were obtained for similar users different from the reference ones (even 100%). These users were computed as similar to the reference ones during the different cluster formations (they also rate the same three or five movies), but they were not included in those clusters.

Table 3: Best right movie recommendations rate (in %) computed using cluster preferences achieved by the GA at the testing phase.

Cluster	10k	15k	20k
276-3M	59.40	59.90	55.40
276-5M	63.96	59.34	65.16
710-3M/5M	61.09	51.80	57.68

Table 4: Right movie recommendations rate (in %) computed using cluster preferences to estimate similar users' ratings on 20 recently rated movies.

Cluster	User	Right	User	Right
276-3M-10k	276	75.00	251	80.00
276-3M-15k	276	75.00	54	90.00
276-3M-20k	276	75.00	554	83.33
276-5M-10k	276	65.00	382	60.00
276-5M-15k	276	75.00	906	100.00
276-5M-20k	276	80.00	879	90.00
710-3M/5M-10k	710	70.00	62	80.00
710-3M/5M-15k	710	70.00	533	75
710-3M/5M-20k	710	70.00	492	70

Finally, the best results for reference users were obtained with the cluster formed for user 276 with 5 starting ratings and 20 neighbors (276-5M-20k). In this case, the obtained attribute weights indicated that the essential movie attributes for users similar to 276

were: release date (20%), action (18.46%), comedy (18.52%) and drama (27.65%) movie genres. The dispensable movie attributes (0%) were animation, childrens, crime, documentary, fantasy, film noir, musical, mystery, war and western movie genres.

4 CONCLUSIONS

This work shows the application of a genetic algorithm for capturing user preferences in Recommender Systems. More specifically, the genetic algorithm allows setting an appropriate weight for each attribute of a product or service, thus defining the preferences of one user or a set of users over these attributes. These obtained attribute weights permit to determine which product or service attributes are more or less valuables for users in their selection process when buying products or paying for services.

All this ranking information given by the GA could be employed by companies which want to offer their clients those products which best fit their desired attributes. Also, as it has been shown in this paper, the obtained weights could be employed to predict user ratings for a test set of movies with a highly good rate of right predictions. Moreover, the GA results could be used for offering good recommendations even when there is not enough information of the current user. In this way, the GA could be employed for capturing the preferences of the users of the system that are similar to the current user, and then for applying the learned weights in order to provide suitable recommendations to the current user. This aspect is especially useful when new users enter into a Recommender System, since the genetic approach permits offering acceptable recommendations even when few starting data for a user is available.

ACKNOWLEDGEMENTS

Thanks to GroupLens Research Group to allow the use of MovieLens data for researching purposes. This work is partially supported by CONSOLIDER-INGENIO 2010 under grant CSD2007-00022 and TIN2008-04446/TIN project, which is co-funded by the Spanish government and FEDER funds.

REFERENCES

Goldberg, D. (1991). Real-coded genetic algorithms, virtual alphabets, and blocking. *Complex Systems*, 5:139–157.

Guan, S., Ngoo, C., and Zhu, F. (2002). Handy broker: an intelligent product-brokering agent for m-commerce applications with user preference tracking. *Electron. Commer. Res. Appl.*, 1(Issues 3-4):314–330.

Herlocker, J. L., Konstan, J. A., Borchers, A., and Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. In *ACM SIGIR '99 Proceedings*, pages 230–237, New York, NY, USA. ACM.

Ortiz, D., Hervas, C., and Muñoz, J. (2001). Genetic algorithm with crossover based on confidence interval as an alternative to traditional nonlinear regression methods. In *ESANN'2001 Proceedings*, pages 193–198, Bruges,Belgium.

Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2000). Analysis of recommendation algorithms for e-commerce. In *ACM EC '00 Proceedings*, pages 158–167, New York, NY, USA. ACM Press.

Sarwar, B. M., Konstan, J. A., Borchers, A., Herlocker, J., Miller, B., and Riedl, J. (1998). Using filtering agents to improve prediction quality in the grouplens research collaborative filtering system. In *CSCW '98 Proceedings*, pages 345–354, New York, NY, USA. ACM Press.

Shardnand, U. and Maes, P. (1995). Social information filetering: Algorithms for automating "word of mouth". In *ACM CHI'95 Proceedings*, pages 210–217.

Shibata, H., Hoshiai, T., Kubota, M., and Teramoto, M. (6-7 Nov. 2002). Agent technology recommending personalized information and its evaluation. In *2nd International Workshop on Autonomous Decentralized System, 2002*, pages 176–183.

Valero, S., Argente, E., Botti, V., Serra, J., and Corma, A. (2004a). A soft computing technique applied to industrial catalysis. In *ECAI2004 Proceedings*, pages 765–769. IOS Press.

Valero, S., Argente, E., Botti, V., Serra, J., and Corma, A. (2004b). Soft computing techniques applied to catalytic reactions. *LNAI*, 3040:550–559.

Valero, S., Argente, E., Botti, V., Serra, J., Serna, P., Moliner, M., and Corma, A. (2009). Doe framework for catalyst development based on soft computing techniques. *Comput. Chem. Eng.*, 33:225–238.