Using Genetic Algorithms for Measuring the Similarity Values between Users in Collaborative Filtering Recommender Systems

Bushra Alhijawi

Hashemite University
Computer Information Systems
Zarqa, Jordan
Email: bushra.hijawi@itc.hu.edu.jo

Yousef Kilani

Hashemite University
Computer Information Systems
Zarqa, Jordan
Email: ymkilani@hu.edu.jo

***Abstract*—Recommender systems aim to help web users to find only close information to their preferences rather than searching through undifferentiated mass of information. Currently, collaborative filtering is probably the most known and commonly used recommendation approach in recommender systems. In this paper, we present a new genetic algorithms-based recommender system, SimGen, that computes the similarity values between users without using any of the well-known similarity metric calculation algorithms like Pearson correlation and vector cosine-based similarity. The results obtained present 46% and 38% improvements in prediction quality and performance, respectively when compared with other techniques.**

Keywords–collaborative filtering; recommender system; genetic algorithms; similarity functions.

I. INTRODUCTION

In everyday life, people rely on recommendations from other people by spoken words, reference letters, news reports from news media, general surveys, travel guides and so forth [1]. Recommender systems (RSs) assist and augment this natural social process to help people sift through available books, articles, web pages, movies, music, restaurants, jokes, grocery products and so forth to find the most interesting and valuable information for them [1]. Currently, the fast increase of Web 2.0 has led to the proliferation of collaborative websites in which the number of elements that can be recommended (e.g blogs) can be increase significantly when introduced (and not voted) by the users. Which generate new challenges for researchers in the field of RSs [2]. RSs are currently being applied in many different domains [3] that can be found in many modern applications such that DVD rental provider Netflix[1], the online book retailer Amazon.com[2] and the prediction of particular stock to the users in the PredictWallStreet company.

The basic principle of RS is the expectation that a group of users similar to one given user, (i.e. the group that have rated an important number of elements in a similar way to the user) can be used to adequately predict that group ratings on products which the user has no knowledge of [2].

Bobadilla et al. [4] categorized RSs into three major groups: collaborative filtering [3], [5], [6], content-based filtering [7] and demographic filtering [8]. Collaborative filtering RSs aggregate ratings or recommendations of objects (e.g. films, books, videos, etc.), recognize commonalities between users on the basis of their ratings and generate new recommendations based on inter-user comparisons [9]. Currently, collaborative filtering is probably the most known and commonly used recommendation approach in RS [10]. In a content-based filtering systems, the objects of interest are defined by their associated features [9] based on user choices made in the past (e.g. in a web-based e-commerce RS, if the user purchased some fiction films in the past, the RS will probably recommend a recent fiction film that he has not yet purchased on this website) [4].

The content-based filtering RS makes recommendations by analyzing the contents like text, images and sounds. In other words, content filtering tools select the right information for the right people by comparing representations of the content contained in the documents to the representations of the content that the user is interested in [11]. The main difference between CF and content-based RSs is that CF only uses the user-item ratings data to make predictions and recommendations, while content-based RSs rely on the features of users and items for predictions [12]. Demographic RSs aim to categorize the user based on personal attributes (i.e. sex, age, county, etc) to make recommendations based on demographic classes [9]. On the other hand, other RSs are hybrid systems that combine two or more groups of the previous three groups. Hybrid filtering is usually based on bio inspired or probabilistic methods such as genetic algorithms (GA), fuzzy genetic, neural networks, Bayesian networks, clustering and latent features [4]. The hybrid user-based fuzzy CF method [13] is an example of hybrid filtering algorithm.

Our interest in this paper is the employment of the genetic algorithms in CF RSs which proposed by [14], [15], [16], [17]). In this paper, we present a novel genetic algorithms-based CF RS, SimGen[3]. The fundamental objective of SimGen is to improve the results, the accuracy and the speed of CF

[1] www.netflix.com
[2] www.amazon.com

[3] The source code can be obtained by emailing one of the authors

978-1-5090-0806-3/16/$31.00 copyright 2016 IEEE
ICIS 2016, June 26-29, 2016, Okayama, Japan

RSs by developing a new similarity metric. SimGen computes the similarity metric using the genetic algorithms and does not use any of the common similarity functions like Pearson correlation and vector cosine-based similarity. Initially, in SimGen, the population consists of a set of members, each represents a similarity metric. Where each member contains a random generated similarity value in the range of [0..1] between every two users, then we split the data for next process in two parts: training data and testing data. The training data is used in order to measure the fitness of the member when selecting the parents for mating after the crossover and mutation operators. The fitness and the testing data is used to test the accuracy of the prediction.

The work presented in this paper makes the following contribution:

- A new mechanism for calculating the similarity values between users using GA is proposed.

The rest of this paper is organized as follows. Section II covers the related work in the RS and GAs. Section III shows the proposed algorithm, SimGen, and the genetic operators that are used. Section IV clarifies the evaluation method that is used and compares SimGen with other techniques. Section V concludes the paper and presents the future work.

II. Related Work

Bobadilla et al. [4] presented a recent survey of the RSs. Su and Khoshgoftaar [1] introduced a comprehensive survey of collaborative filtering techniques. In this section, we review the related work on the RS using GAs.

Bobadilla et al. [2] developed a genetic algorithms-based RS, GARS, that is based on CF which use the rating of users to compute the similarity. Bobadilla et al. [2] built a new similarity function that measures the similarity between every two users. GARS makes statistics about every two users. It calculates how many times both users rate items exactly with a difference of 1, 2, 3, and 4. The later calculated value represents the difference between the rating of two users. GARS associate a weight with each value then it computes the weight by using the GAs.

Gao and Li [16] proposed a hybrid model to integrate outputs produced by every RS at the basis of GAs by constructing weight vectors that represent different forecasting performances of each RS. The hybrid problem is translated into optimizing problem about weight vectors. Fong et al. [17] designed a recommender that exploits advantages of hybrid CF for high quality prediction and recommendation. They presented a GA-based approach for supporting combined modes of CF to recommend new items for a particular user based on his previous likes or the opinions of other like-minded users. Salehi et al. [18] proposed a hybrid RS for learning materials based on their attributes. This system consists of two modules process. The weights of implicit or latent attributes of materials for learner are used as chromosomes in genetic algorithms. Then it optimizes the weights according to the historical rating. Al-Shamri and Bharadwaj [19] developed a hybrid fuzzy-genetic RS by employing GA to evolve appropriate weights for each feature of the user model. In addition, they proposed a novel fuzzy distance metric to match users and user model that enables hybrid filtering which reduces the system and computational time.

Jia et al. [20] designed a RS that uses the genetic algorithms. It selects weights and threshold values to be used in the similarity calculation. This system selects neighbors based on the similarity which then gives the recommendation based on trust. It considers the similarity and trust as necessary and considers the characteristics of the users and the items own characteristics to improve the effect. Finally, the system gets the more accurate selection of the best neighbors to make a further improvement of the recommendation accuracy.

All the reviewed papers use a similarity metric in order to calculate the similarity between users or items. This similarity metric is either one of the well-known matrices or it is based on calculating the statics or historical data about the rating of the users. Our algorithm, SimGen, does not use any similarity function. It starts by giving a random initial similarity value between every two users and this similarity value is enhanced from generation to generation as SimGen runs. This enhancement is done by the direction of the training data, and SimGen does not require any additional information provided by hybrid model like Bobadilla et al. [2], [20].

III. Using GA to Compute the Metric

The main objective is to improve the results of CF by using GA to compute the similarity metric. SimGen does not use any of the common similarity functions like Pearson correlation and vector cosine-based similarity. It generates an initial random similarity value between every two users then it runs the GA. The proposed algorithm uses training data to compute the fitness in every generation and the testing data to test the resulting similarity to compute the errors in prediction.

A. Initial Population

The initial population consists of a set of individuals. We represent each individual by a 2-dimensional array as shown in fig. 1. Initially, we generate a random similarity value in the range [0..1] between every two users for each individual. It is a common knowledge that the similarity values in RS ranges between [-1..1] and the similarity value between the user and itself is 1.

		User				
		1	2	3	...	x
User	1	1				
	2	sim(2,1)	1			
	3	sim(3,1)	sim(3,2)	1		
	...	sim(...,1)	sim(...,2)	sim(...,3)	1	
	x	sim(x,1)	sim(x,2)	sim(x,3)	sim(x,...)	1

Figure 1. similarity matrix between users

B. Fitness Function

The fitness function is a major component of GA that is used to evaluate the individuals in the population. In SimGen, the role of the fitness function is to measure the optimality of the individual (i.e. similarity array).

In order to calculate the fitness function for each individual, we use the similarity metric (i.e the individual) to compute the prediction of each user rated items for each training user. The prediction of user x on item i is given by the following well-known formula:

$$p_x^i = \bar{r}_x + \frac{\sum_{n=1}^{k_x}[sim(x,n) * (r_n^i - \bar{r}_n)]}{\sum_{n=1}^{k_x} sim(x,n)} \quad (1)$$

TABLE I. SIMGEN COMPONENTS

Initialization	Initial population	The initial population consists of 50 individuals, each one contains a random similarity value in range [0..1] between every two users.
Assessment	Fitness Function	The Mean Absolute Error (MAE) is computed depending on the predicted rates of items that are rated by each of the training users.
Genetic Operators	Selection	Roulette-Wheel selection (RW) technique is used.
	Crossover	Uniform crossover operator is used with 50% probability.
	Mutation	Single point mutation technique with 20% probability.

Where,

- $\bar{r}_x$ represents the average ratings made by user x for the training items x rated.
- $sim(x, n)$ is the similarity value between users x and n. This value is taken from the individual.
- r_n^i represents the rating of user n on item i.
- k_x is the set of the neighbor users to user x. In SimGen, we consider k_x to be the set of the training users (80% of the all the users) which x rated.

The MAE (equation 2) is obtained by computing the difference of the actual ratings with the predicted ratings that are produced depending on the similarity array. We calculate the predictions for all the training users on all the training items based on the similarity array in order to compute the MAE of the RS. The lowest fitness value for individual z means that individual z provides closest predictions. The objective of running SimGen is to obtain an individual with the smallest MAE. Therefore, the fitness value for a member m is better than the fitness value for a member n means that m is a better similarity metric than n since the users have predicted values according to the similarity metric in m closer to their actual rate than those in n. Consider the following example which illustrates the idea.

$$MAE = \frac{1}{\#U} \sum_{u=1}^{U} \frac{\sum_{i=1}^{I_u} |p_u^i - r_u^i|}{\#I_u} \quad (2)$$

Where, $\#U$ represents the number of training users and $\#I_u$ represent the number of training items rated by the user u.

Example 1. Suppose that there are 5 users, 10 items, and each of the 5 users rated at least three items as shown in fig. 2 and there are two members m and n as shown in fig. 3. It is clear that m is better than n since the accuracy of prediction resulted from m is higher than resulted from n as shown in fig. 4. The MAE (equation 2) result from using the similarity matrix m is 0.958 while it is 1.3 when use the similarity matrix n. $\diamond$

	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}
U_1	3	.	5	1	1	.	2	.	.	.
U_2	1	2	.	3	.	.	.	.	.	.
U_3	.	.	4	.	.	1	.	3	5	.
U_4	.	2	.	2	.	.	5	3	.	4
U_5	4	.	4	2	2	3	.	.	3	.

Figure 2. User-Item rating matrix

C. Genetic Operators

SimGen uses the selection, crossover and mutation operators. The crossover and mutation probability is 0.5 and 0.2 respectively. We select the individuals for mating based on their fitness value. As a selection technique, the roulette-wheel parent selection algorithm(RW) is used to select the parents for mating. RW gives high probability value for the good individuals and low probability value for the bad individuals to be selected. Note that both the good and the bad individuals have chances to be selected, however the good individuals have higher chances. The uniform crossover operators is applied and two new children are generated after each round of crossovers. These children replace the worst two members in the current population in case they have better fitness. Otherwise, they are discarded. We apply the mutation operation on the children produced as a result of the crossover operation with a probability of 0.2.

Member M

	U_1	U_2	U_3	U_4	U_5
U_1	1				
U_2	0	1			
U_3	0.3	0	1		
U_4	0.11	0.7	0.2	1	
U_5	0.71	0.2	0.8	0.2	1

Member N

	U_1	U_2	U_3	U_4	U_5
U_1	1				
U_2	0.4	1			
U_3	0	0.3	1		
U_4	0	0	0.5	1	
U_5	0.2	0.5	0.3	0.2	1

Figure 3. Members M and N

Prediction Rates M

	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}
U_1	3.4	.	3.4	1.3	1.4	.	1.2	.	.	.
U_2	3	1.8	.	1.8	.	.	.	.	.	.
U_3	.	.	3.6	.	.	3.25	.	3.1	3.25	.
U_4	.	2.2	.	2.8	.	.	2.8	2.9	.	3.2
U_5	3	.	4.6	1.9	1.6	0.75	.	.	4.75	.

Prediction Rates N

	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}
U_1	1.4	.	3.4	2.7	1.4	.	2.4	.	.	.
U_2	3.8	2.4	.	1.8	.	.	.	.	.	.
U_3	.	.	4.25	.	.	3.25	.	3.45	3.25	.
U_4	.	3.2	.	2.2	.	.	3.2	2.95	.	3.2
U_5	2.4	.	3.3	2.4	1.6	0.75	.	.	4.75	.

Figure 4. The prediction rates of members M and N

IV. EXPERIMENT

In this section, we experimentally prove that the prediction quality and the performance of our algorithm, SimGen, is better than the results obtained by other techniques like cosine metric, Pearson correlation, and genetic-based algorithms (i.e. GARS [2]). We have used in the experiments two different datasets:

- Movielens. Movielens [4] database. The Movielens presents a common reference in RS research that consists of 100,000 ratings (1-5) from 943 users on 1682 movies. Each user has rated at least 20 movies.
- Synthetic data. Which is a random biased data as Marmanis and Babenko [21] did to test their recommendation engine which they called Delphi. This data consists of 100 users, u_1, u_2,..., u_{100}, 300 movies,

[4] downloadable from http://grouplens.org/datasets/movielens/

i_1, i_2,..., i_{300}. The users and items are splitted into sets as follows:

- $U_{40} = \{u_1,..., u_{40}\}$, contains 40% of the total users.
- $U_{60} = \{u_{41},..., u_{100}\}$, contains the other 60% of the users.
- $I_{40} = \{u_1,..., u_{40}\}$, contains 40% of the total items.
- $I_{60} = \{u_{41},..., u_{100}\}$, contains the other 60% of the items.

Three constraints were placed to build this data:

1) Each user has rated n movies, where n is a random number between 10 and 20.
2) n items from I_{40} assigned randomly to each user from U_{40}, a random rate set to 4 or 5.
3) n items from I_{60} assigned randomly to each user from U_{60}, a random rate set to 1, 2 or 3.

The synthetic random data built in a such a way, can be easily determined if the result of running any algorithm using it is correct answer. For instance, the prediction of any user from U_{60} for any item of I_{60} should be near (1, 2 or 3). Otherwise, the prediction is not accurate.

Table II shows the main parameter used in the experiments. Each dataset was divided into training data (i.e. 80%) and testing data (i.e. 20%). All the algorithms were built using Visual Basic programming language and the CPU time is measured using Process Explorer software. We used a PC of 2 GB RAM with Dual CPU 2.16 GHz.

TABLE II. MAIN PARAMETERS USED IN THE EXPERIMENTS

Dataset	Precision/Recall N	K-Neighbors	Test user %	Test item %	Genetic runs
Movielens	{2,...,20}	{20,40,60,...,200}	20%	20%	100
Synthetic data	{2,...,20}	{2,4,6,...,20}	20%	20%	100

The results of $SimGen$ are compared with the ones obtained from the other approaches using traditional metric on CF RS: Mean Absolute Error (MAE), precision, and recall. Each of the 4 techniques: $SimGen$, Bobadilla et al. [2], Pearson correlation and cosine metric were ran for 100 times and the average of the 100 runs is taken.

Table III present the average CPU time in seconds was taken to make the prediction by the 4 algorithms. It is clear from the results that that there is a high improvement in speed using SimGen. SimGen takes 15.8 while GARS [2] takes 22.65 seconds, cosine metric takes 27.6 seconds, and Pearson correlation takes 25.9 seconds. This means that SimGen is one and half times (22.65/15.8 = 1.5, 25.9/15.8 = 1.6) faster than GARS [2] and Pearson correlation, and nearly two times (27.6/15.8=1.7) faster than cosine metric.

TABLE III. THE PERCENTAGE OF IMPROVEMENT IN SPEED RESULTING FROM SIMGEN

	SimGen	GARS [2]	Cosine metric	Pearson correlation
CPU time	15.8s	22.65s	27.6s	25.9s
Improvement = $\frac{algorithm(A)-SimGen}{algorithm(A)}$		30.2%	42.79%	39%

Figs. 5, 6 and 7 show the results of the MAE, recall, and precision measures of the 4 techniques. The number of neighbor users for a user is represented by the constant k in fig. 5. These neighbors vary between 20 to 200 and 2 to 20 when using Movielens and synthetic data respectively.

The results of recall and precision metrics show in figs. 6 and 7 respectively. Fig. 6 presents that the quality is improved in any number of recommendation. Also in fig. 7, it is clear that the quality measure is improved for any value used in the number of recommendation using both datasets.

V. CONCLUSION

In this paper, we presented a new genetic-based algorithm, SimGen, for computing the similarity metric. We showed that SimGen does not use any of the existing similarity functions like Pearson correlation and vector cosine-based similarity. In addition, it does not require the additional information provided by the hybrid models. Initially, SimGen gives the similarity between every two users a random value. Then, it runs to adjust this value based on the training data.

In order to evaluate SimGen, the traditional metric on CF RS: MAE, precision, and recall. We proved that SimGen achieved high performance in accuracy and quality compared to the current state-of-the-art genetic-based algorithms like GARS [2], cosine metric, and Pearson correlation. In the compression process, two data sets were used: synthetic data and Movielens. The results obtained present that SimGen is:

- Using synthetic data:
  1) One and half times faster than GARS [2] and Pearson correlation.
  2) Two times faster than cosine metric.
  3) 33% improvement in prediction quality than GARS [2].
  4) 69.5% improvement in prediction quality than cosine metric.
  5) 55.5% improvement in prediction quality than Pearson correlation.

- Using Movielens data:
  1) One and half times faster than GARS [2] and Pearson correlation.
  2) Two times faster than cosine metric.
  3) 14.66% improvement in prediction quality than GARS [2].
  4) 70% improvement in prediction quality than cosine metric.
  5) 46.5% improvement in prediction quality than Pearson correlation.

REFERENCES

[1] X. Su and T. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques," Journal of Advances in Artificial Intelligence, 2009.

[2] J. Bobadilla, A. Ortega, F.and Hernando, and J. Alcalá, "Improving Collaborative Filtering Fecommender System Results and Performance using genetic algorithms," Journal of Knowledge-Based Systems, vol. 24, 2011, pp. 1310–1316.

[3] P. Moradi and S. Ahmadian, "A Reliability-based Recommendation Method to Improve Trust-Aware Recommender Systems," Journal of Expert Systems with Applications, vol. 42, no. 21, 2015, pp. 109–132.

[4] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender Systems Survey," Journal of Knowledge-Based Systems, vol. 46, no. 8, 2013, pp. 109–132.

[5] S. Ghazarian and M. Nematbakhsh, "Enhancing Memory-based Collaborative Filtering for Group Recommender Systems," Journal of Expert Systems with Applications, vol. 42, no. 7, 2015, pp. 3801–3812.

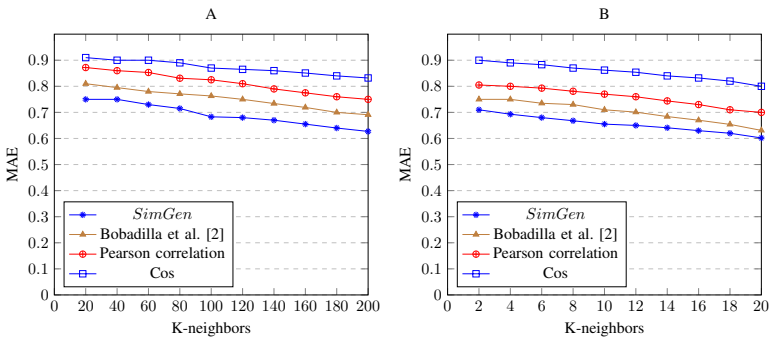

Figure 5. The MAE measure of the Bobadilla et al. [2], the Pearson correlation, the cosine metric, and the $SimGen$ algorithms. A) Movielens. B)Synthetic data.

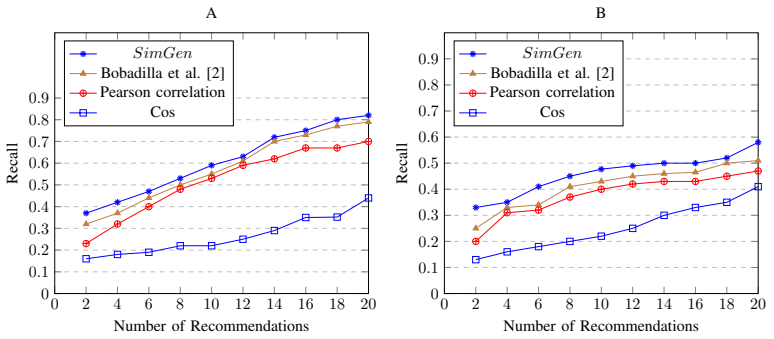

Figure 6. The recall quality measure of the Bobadilla et al. [2], the Pearson correlation, the cosine metric, and the $SimGen$ algorithms. A) Movielens. B)Synthetic data

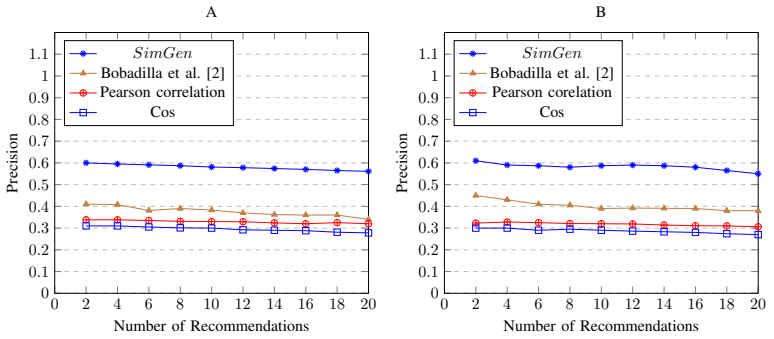

Figure 7. The precision quality measure of the Bobadilla et al. [2], the Pearson corelation, the cosine metric, and the $SimGen$ algorithms. A) Movielens. B)Synthetic data

[6] K. Christidis and G. Mentzas, "A Topic-based Recommender System for Electronic Marketplace Platforms," Journal of Expert Systems with Applications, vol. 40, no. 11, 2013, pp. 4370–4379.

[7] G. Adomavicius and A. Tuzhilin, "Context-Aware Recommender Systems," Recommender Systems Handbook, F. Ricci, et al. (Ed.), 2011, pp. 217–253.

[8] I. Ma, H.and King and M.-R. Lyu, "Learning to Recommend with Explicit and Implicit Social Relations," ACM Transactions on Intelligent Systems and Technology, vol. 2, no. 3, 2011.

[9] R. Burke, "Hybrid Recommender Systems: Survey and Experiments," Journal of User Modeling and User-Adapted Interaction, 2002, pp. 331–370.

[10] Q. Shambour and J. Lu, "A Hybrid multi-criteria semantic-enhanced collaborative filtering approach for personalized recommendations," IEEE WICACM International Conference on Web Intelligence, 2001.

[11] J. Herlocker, J. Konstan, A. Borchers, and J. Riedl, "An Algorithmic Framework for Performing Collaborative Filtering," In: 22^{nd} annual international ACM SIGIR conference on Research and development in information retrieval, 1999, pp. 230–237.

[12] L. Si and R. Jin, "Flexible Mixture Model for Collaborative Filtering," In: the 20^{th} International Conference on Machine Learning, vol. 2, 2003, pp. 704–711.

[13] L. Son, "HU-FCF: A Hybrid User-based Fuzzy Collaborative Filtering Method in Recommender Systems," Journal of Expert Systems with Applications, vol. 41, no. 15, 2014, pp. 6861–6870.

[14] K. Kim and H. Ahn, "A Recommender System Using GA K-means Clustering in an Online Shopping Market," Journal of Expert Systems with Applications, vol. 34, no. 2, 2008, pp. 1200–1209.

[15] ——, "Using a Clustering Genetic Algorithm to Support Customer Segmentation for Personalized Recommender Systems," In: 13^{th} International Conference on AI, Simulation, and Planning in High Autonomy Systems, 2005, pp. 409–415.

[16] L. Gao and C. Li, "Hybrid Personalized Recommended Model Based on Genetic Algorithm," In: 4^{th} International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM), 2008, pp. 1–4.

[17] S. Fong, Y. Ho, and Y. Hang, "Using Genetic Algorithm for Hybrid Modes of Collaborative Filtering in Online Recommenders," In: 8^{th} International Conference on Hybrid Intelligent Systems (HIS), 2008, pp. 174–179.

[18] M. Salehi, M. Pourzaferani, and S. Razavi, "Hybrid Attribute-based Recommender System for Learning Material using Genetic Algorithm and a Multidimensional Information Model," Egyptian Informatics Journal, vol. 24, no. 1, 2013, pp. 67–78.

[19] M. Al-Shamri and K. Bharadwaj, "Fuzzy-Genetic Approach to Recommender Systems based on a Novel Hybrid User Model," Journal of Expert Systems with Applications, vol. 35, no. 3, 2008, pp. 1386–1399.

[20] Y. Jia, Q. Ding, D. Liu, J. Zhang, and Y. Zhang, "Collaborative Filtering Recommendation Technology based on Genetic Algorithm," Journal of Applied Mechanics and Materials, vol. 599–601, no. 3, 2014, pp. 1446–1452.

[21] H. Marmanis and D. Babenko, Algorithms of the intelligent web minning, 2^{nd} ed., 2009.