International Journal of Engineering & Technology, 7 (4.33) (2018) 5-9

SPC

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper



An Evaluation Framework of Trust Aware Recommender System

S. Masrom[1]*, N. Khairuddin[1], A. Abdul Rahman[1], A. Azizan[1], A.S.A. Rahman[2]

[1]Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Perak Branch, Tapah Campus, Tapah Road, 35400 Perak, Malaysia
[2]Faculty of Science and Information Technology, Universiti Teknologi PETRONAS, Seri Iskandar, 32610 Perak, Malaysia
**Corresponding author E-mail: suray078@perak.uitm.edu.my*

Abstract

To date, there exists a variety of prediction approaches have been used in recommender systems. Among the widely known approaches are Content Based Filtering (CBF) and Collaborative Filtering (CF). Based on literatures, CF with users rating element has been widely used but the approach faced two common problems namely cold start and sparsity. As an alternative, Trust Aware Recommender Systems (TARS) for the CF based users rating has been introduced. The research progress on TARS improvement is found to be rapidly progressing but lacking in the algorithm evaluation has been started to appear. Many researchers that introduced their new TARS approach provides different evaluation of users' views for the TARS performances. As a result, the performances of different TARS from different publications are not comparable and difficult to be analyzed. Therefore, this paper is written with objective to provide common group of the users' views based on trusted users in TARS. Then, this paper demonstrates a comparison study between different TARS techniques with the identified common groups by means of the accuracy error, rating and users coverage. The results therefore provide a relative comparison between different TARS.

Keywords: *Collaborative filtering; Trust aware; Recommender system; Evaluation framework.*

1. Introduction

Since the last decades, the cumulative progress of knowledge and information from the Internet technology has been tremendous. With the wide application of the technology, the peoples' ability to process the beneficial information is relatively crucial to everyone. To provide this beneficial information quickly from the huge repository of web and mobile applications, recommender systems have appeared and has gained wide attention from the community. Today, recommender systems have a significant impact to the way of peoples finding the best products, information and even other peoples and contacts.

Recommender system plays a role to filter large information search space and to select the most suitable items that are likely to be more interesting and attractive to a user. Recommender systems have been proved to be beneficial used in many kinds of application domain such as online job directories, online libraries, e-commerce and social networks, including Facebook and LinkedIn. Besides that, with the rapid development of e-commerce, recommender systems have been utilized as an important tool for the seller and customers.

Since the introduction, there exists many approaches have been used to implement a recommender system. One of the popular approaches is Collaborative Filtering (CF) that utilizing user ratings based on items [1]. Another approach is Content-Based Filtering (CBF) that uses content information of items in measuring the matching values between the items and users[2]. Additionally, demographic information, such as, age, gender and occupation, in the user profile have also been used to recommend items to the users [3-4]. Although recommender system has been widely used, some crucial problems remain appeared in the implementation for examples cold start and sparsity problems. Cold start problem appears due to the existence of new users or items that not received any ratings [5]. Furthermore, if the number of rating on the existing items is very small, the sparsity problem occurs. As the number of items is rapidly increasing while the users rating is progressively slow, the cold start and sparsity problems would create less rating coverage and inaccurate recommendations [3]. In order solve the problems, a recommender system with trust aware elements have been introduced [6-7]. The recommender system is called as Trust Aware Recommender System (TARS).

It has been reported by many researchers that the accuracy of TARS is better than the traditional CF approach [7-8]. Since the introduction in the early of 2012, different techniques of TARS have been introduced to improve the cold start and sparsity problems. One of the current techniques is TARS with distrust element [3]. It is anticipated in this research that further performance observation of TARS with distrust should be conducted as the technique is still new and current evaluation has been difficult to be compared due to less structured of performances evaluation. In this research, the interest has been directed to observe the performances of TARS with distrust in different types of trusted users' views. The types of trusted users' view are priory identified based on previous studies.

This paper is organized as follow. The next part provides research background of recommender systems and TARS followed with research methodology in part 3. Part 4 reports the results and discussions before the concluding remarks at part 5.

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

2. Research Background

This part describes different approaches of recommender system, including trust aware recommender systems.

2.1. Recommender System (RS)

Two common approaches for RS are Collaborative Filtering (CF) and Contents-based Filtering (CBF). CF recommender system utilizing a group of users' information and also the attribute similarity of the related items [9]. In other words, it provides recommendations to a user that are based on recommendations given by other users with similar interest or profiles. In other words, it considers the ratings provided by the related users. CF has been widely used by the majority e-commerce systems like Amazon, Lazada and Facebook. To date, CF can be classified into two sub-categories namely memory-based and model-based [8]. *Memory-based approaches* make predictions by utilizing the ratings from the active users, which stored in the memory caches of the users' devices [9-10]. Conversely, *model-based approaches* utilizing the construction of training model based on classification and clustering paradigms. Then, it will make prediction based on the training model onto another set of real [11]. Typical examples that used this approach are clustering models [12-13] and machine learning [14]. Computational intelligent approaches like fuzzy [15] and meta-heuristics algorithms [16-17] are also popular for these applications. Contrast with CF, Content-based Filtering Systems (CBF) recommender system utilizing information receives from the active users and data about the items associated. It makes recommendations by comparing the users' profiles that consist the content of document collections. The technique focuses more on the characteristics of the users and item rather than utilizing other data such user rating [3, 8]. Without users rating inclusion, the technique has an advantage in recommending more accurate contents to users [2]. The extensively used of CF in recommender system has given an attraction for this research to focus on the Trust Aware Recommender System (TARS).

2.2. Trust Aware Recommender System (TARS)

Trust-Aware Recommender System (TARS) is basically the consequence of traditional CF approach. TARS reflects to trust link between users in order to generate recommendations [6]. Research has proved that TARS can efficiently overcome data sparsity and cold start problems, which appeared in the traditional CF approaches.
The technique for TARS is common to the traditional CF. If the weight of each recommendation in traditional CF counted an active user similarity, TARS in [7] allow the inclusion of active user trust recommendation, which consists of two steps. The first step is the trust measurement calculated by trust metric used as a weight parameter of the second step to replace the user similarity weight. The second step is defined as in (1).

$$p_{a,i} = \bar{r}_a + \frac{\sum_{u=1}^{k} w_{a,u}\left(r_{u,i} - \bar{r}_u\right)}{\sum_{u=1}^{k} w_{a,u}} \tag{1}$$

where $p_{a,i}$ presents the predicted rating what an active user a would possibly provide for item $i, \bar{r}_a$ is the average rating values given by the active user, k is the number of users who ratings the item i. Then, $r_{u,i}$ is the rating value of user u to item $i, \bar{r}_u$ is the average of the rating values provided by user u to item i, $w_{a,u}$ is the user similarity weight of a and u as computed in (2).

$$w_{a,u} = \frac{d_{\max} - d_{a,u} + 1}{d_{\max}} \tag{2}$$

where $d_{\max}$ is the maximum allowable propagation distance (MAPD) between users of the recommender system. The value of MAPD can be preset. Then, $d_{a,u}$ is the active user a trust propagation distance to the recommender u. In TARS, the trust propagation distance refers to the number of hops in the shortest trust propagation path from the truster to the trustee. As in (1) and (2) used a measure of active user *a* to the recommender *u*, the network trust property is a kind of local network. The trust values are directly provided by user *u* to user *a*, therefore the trust establishment is explicit.
Furthermore, researchers in [17] proposed new formulation of TARS that extended the basic Epinions dataset with distrust statement. The formula of calculating $w_{a,u}$ in (2) has been changed as denoted in (3).

$$w_{a,u} = T_{a,u} - d_{a,u} \tag{3}$$

In (3) decreases the amount of propagated distrust from propagated trust of user against to the users that gives rating on common items *i*, where

$$T_{a,u} = \frac{d_{\max} - d_{a,u} + 1}{d_{\max}}, D_{a,u} = \frac{dt_{\max} - dt_{a,u} + 1}{dt_{\max}} \tag{4}$$

where, $T_{a,u}$ refers to amount of propagated trust d from user $'a'$ to user $'u'$ and $D_{a,u}$ calculates the distrust values based on the propagated distrust dt from user $'a'$ to user $'u'$. While, $d_{\max}$ approximately equals to the average path length of trust network's corresponding random network.

$$d_{\max} = \left[L^R\right] = \left[\frac{\ln n}{\ln k}\right] \tag{5}$$

In (5), L^R is the average path length of the network in the corresponding random network related to trust network, n is the size of trust network, and k is average of trust network degree. Empirical experiments have been conducted by researchers in [7] that observed the performances of TARS with in (1) on different sets of views from the Epinions dataset. The results from the experiments have shown a significant impact of the different views in relation to the different tested algorithms.
An interesting research in [18] focuses on computing the predicted rating using a simple version of Resnick's prediction formula based on a single user as denoted in (6).

$$p_{a,i} = \bar{r}_a + \left(r_{u,i} - \bar{r}_u\right) \tag{6}$$

where $\bar{r}_a$ and $\bar{r}_u$ refer to the mean ratings of users a and u respectively. $r_{u,i}$ is the rating of item i given by user u. The trust score is then derived by averaging the prediction error on co-rated items as follows in (7).

$$t_{a,u} = \frac{1}{\left|I_{a,u}\right|} \sum i \xi I_{a,u} \left(1 - \frac{p_{a,i} - r_{a,i}}{r_{\max}}\right) \tag{7}$$

where $I_{a,u}$ refer to the set of rated items of a and u, respectively, and $r_{\max}$ is the size of the rating range. The results prove that the

used of trust into CF can certainly improve the prediction accuracy while maintain the fair prediction coverage. Besides that, another research by [5] also adopt Resnick's prediction formula but compute trust based on mean squared distance (MSD) as shown in (8).

$$t_{a,u} = \frac{\left|I_{a,u}\right|}{\left|I_a \cup I_u\right|}\left(1 - \frac{1}{\left|I_{a,u}\right|}\sum i\xi I_{a,u}\left(1 - \frac{p_{a,i} - r_{a,i}}{r_{\max}}\right)2\right) \quad (8)$$

The users whose trust value is greater than a threshold λ, i.e., $t_{a,u} > \lambda$ are regarded as trusted neighbours. The approach proved to resolve the cold start and sparsity significantly, but there is problem with the computational cost because it is very expensive.
The following part describe the methodology used in this research that compare these TARS techniques. Due to the computational cost, the technique that introduced by [5] is not included in this study.

3. Methodology

This part describes the methodology used by means of performance metrics, parameters, TARS techniques and dataset.

3.1. Performance Metrics

Different performance metrics have been used to evaluate the quality of recommendations in recommender systems. The most common performance metrics used are the standard Mean Absolute Error (MAE) and the Coverage metrics. The MAE is the most widely used metric in recommendation research to measure the accuracy of the recommendations [19]. MAE measures the accuracy by computing the average absolute deviation between the difference between the predicted rating and actual rating assigned by the user. The lower the MAE, the more accurate the predictions are, allowing for better recommendations to be formulated. The formula to calculate MAE is denoted in (9).

$$MAE_i = \frac{\sum_{j=1}^{n_i}\left|ar_{ij} - r_{ij}\right|}{n_i} \quad (9)$$

where ar_{ij} is the real rating related to user i and item j, and r_{ij} is the predicted corresponding rate of user i to item j. While, n is the number of predicted ratings. The rating coverage of TARS is measured by using the following formula in (10).

$$coverage = \frac{n_r}{n_c} \quad (10)$$

where n_r is the total number of items that the recommender system could predict and n_c is the total number of items. Furthermore, Mean Absolute User Error (MAUE) are defined by [6] that calculates the mean error of each user from the MEA of all users. Lastly, user coverage is related with the percentage of users that a recommender system can provide predictions. To summarize, the performance metrics used in this research are MAE, MAUE, rating coverage and users coverage.

3.1. Selected TARS

Three techniques of TARS described in the literature review part are as follow:

- Basic TARS [6]
- Distrust TARS [7]
- Resnick TARS [18]

The MAE and rating coverage of these TARS will be compared according to different types of trust users view described in the following.

3.2. Different Types of Trust Users View

The experiments focused on five views of trusted user as listed in the following Table 1.

Table 1: Five views of trusted users

Views	Characteristics
All users	All types of user
Cold start user	Users who gives 1-4 times ratings
Heavy user	Users who give more than 10 times ratings
Opinionated user	Users who gives 1-4 times ratings and standard deviation of rating value is more than 1.5
Flexible user	Users who give more than 10 times ratings and standard deviation of rating value is more than 1.5

3.3. Epinions Dataset

This research used Epinions and extended Epinions datasets. Epinions.com is a web site that allows users to review various items (cars, books, music, etc.). Figure 1 shows the data representation for the Epinions dataset by [7]. Epinions dataset is divided into two set of datasets that is the basic Epinions dataset and the extended Epinions dataset. Basic Epinions Dataset: The basic Epinions dataset contains 49,290 users who rated a total of 139,738 different items at least once, writing 664,824 reviews and 487,181 issued trust statements. The dataset consists of 2 files: *rating_data* and *trust_data*.

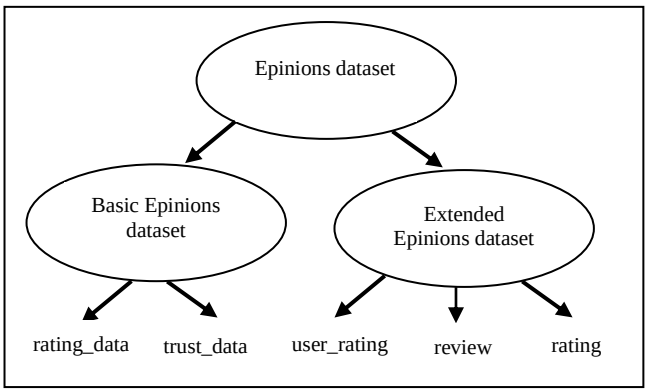

Fig. 1: Epinions Dataset representations

The ratings_data contains the ratings given by users to items for example 1, 2, 3 and 5. Every line in the file of *ratings_data* has the following data, which are *user_id*, *item_id* and *rating_value*. The ranges of the *user_id* is from 1to 49290, *item_id* from 1 to 139738 and *rating_value* from 1 to 5. Figure 2 presents the diagram to illustrate the connection of user_id that gives rating_value to item_id in the rating_data file.

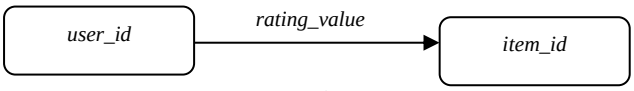

Fig. 2: Rating_data file representation

As for example, the dataset is saved as the following Table 2.

Table 2: Rating_data dataset example

user_id	Item-id	Rating_value
1	101	4
1	102	5
1	103	3
1	104	3

Furthermore, the second file named as trust_data contains trust statements issued by users. Every line in the file has the following data namely *source_user_id*, *target_user_id* and *trust_statement*_value. The ranges of the *source_user_id* and *target_user_id* is from 1 to 49290. Besides, the *trust_statement_value* is always 1 (since in the dataset there are only positive trust statements and not negative ones (distrust)). Figure 3 presents the diagram to illustrate the connection of *source_user_id* that gives *trust_statement_value* to target_user_id in the *trust_data* file.

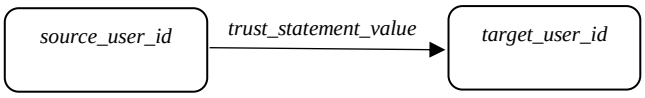

Fig. 3: *Trust_data* file representation

As for example, the dataset is saved in the file as the following Table 3.

Table 3: Trust_data dataset example

source_user_id	target_user_id	trust_statement_value
22605	42915	1
22514	11369	1
30152	44255	1
30152	25278	1

Extended Epinions Dataset: The extended dataset contains also the distrust lists that show which users are distrusted by which users. The dataset contains 132, 000 users, who issued 841,372 statements that include 717,667 trusts and 123,705 distrusts and 85,000 users received at least one statement. The dataset consists of 3 files: *user_rating*, *review* and *rating*.

The *user_rating* file contains the trust and distrust value of the user. It stores *source_user_id* that is the member who is making the trust and distrust statement, target_*user_id* of the member being trusted or distrusted and trust_statement_value that is the value of trust whether the value equal to 1 for trust and -1 for distrust. The dataset is saved such as in the Table 4.

Table 4: User_rating dataset example

source_user_id	target_user_id	trust_statement_value
3287060356	232085	-1
3288305540	709420	1
3290337156	204418	-1
3294138244	269243	1

Furthermore, the review file contains the information about each review that is written by a user. This file consists of three columns that include the following details, the *object_review_id*, *user_review_id* and *review_id*. The dataset is saved as the following in Table 5.

Table 5: Review dataset example

object_review_id	user_review_id	review_id
3287060356	232085	149002163073
3288305540	709420	149002425217
3290337156	204418	5303145344
3294138244	269243	192620893057

The last file is the rating file. This file contains columns of details that include the *item_id*, which is the object that is being rated, *user_id* of the member who is rating the object and *rating_value* from 1 to 5. Value 1 means not helpful, value means 2 somewhat helpful, value 3 means helpful, value 4 means very helpful and value 5 means most helpful. Next, is the *rate_status*, with 1 means the member has chosen not to show his rating of the object and 0 is the member does not mind showing his name besides the rating. The dataset is saved as the following in Table 6.

Table 6: Rating dataset example

item_id	user_id	rating_value	rate_status
139431556	237911	5	0
139431556	409066	2	0
41332100	241261	5	0
143101572	264696	4	1

Many researches on recommender used Epinions dataset because it is the largest and the most significant dataset. In addition, the data has been collected through real world data.

4. Results and Discussion

The results are presented in the following Table 7. The MEA and rating coverage of the three TARS were compared according to five different view namely all users, cold start users, heavy users, opinionated users and flexible users.

Table 7: Accuracy (MAE) and rating coverage measures for different TARS algorithms on different views

Mean Absolute Error (MAE), Ratings Coverage			
Views	TARS		
	Basic	Distrust	Resnick
All users	0.844, 61.8%	0.705,64.11%	0.878, 58.5%
Cold start users	1.099, 3.43%	1.032, 4.52%	1.085, 3.15%
Heavy users	0.862,57.47%	0.728, 77.11%	0.837,57.23%
Opinionated users	1.220, 51.30%	1.165, 58.12%	1.208, 51.01%
Flexible users	0.884, 60.29%	0.738, 81.56%	0.833, 60.67%

In terms of MAE that presents the accuracy of algorithms, the highest accuracy of all TARS has been produced when involving all the users. However, TARS with Distrust technique that seems to be able to improve the accuracy results from basic TARS and Resnick TARS at all views. No matters on what variation of rating values, less number of ratings from cold start and opinionated users have significantly reduced the accuracy of all of the TARS. It can be seen in the table that all the MAE results from cold start and opinionated of All TARS were bigger than the all, heavy and flexible users.

However, with variation of more than 1.5 rating values, the rating coverage can be extremely increased. As shown in the Table 8, rating coverage from cold start users is only 3.43% for Basic TARS, 4.52% for Distrust TARS and 3.15% for Resnick TARS. A great improvement can be seen in the rating coverage from opinionated users, which are 51.3%, 58.12% and 51.01% for all TARS respectively. Besides MAE, the widest coverage also generated from the flexible users.

Furthermore, a comparison of MAUE and users coverage of all TARS in different views is presented as listed in Table 8.

Table 8: Accuracy (MAUE) and users coverage measures for different TARS algorithms on different views

Mean Absolute User Error (MAUE), Users Coverage			
Views	TARS		
	Basic	Distrust	Resnick
All users	0.877, 58.3%	0.721,59.66%	0.892, 57.9%
Cold start users	1.267, 5.85%	1.067, 7.14%	1.228, 5.23%
Heavy Users	0.902, 68.55%	0.882, 61.67%	0.962,67.62%
Opinionated users	1.471, 88.25%	1.361, 71.16%	1.485, 78.85%
Flexible users	0.927, 45.63%	0.845, 38.33%	0.971, 42.33%

Over all views, the MAUE achieved by the three TARS in all users' views seems to be the lowest and TARS with distrust has the smallest MAUE (0.721). Besides, the Distrust TARS has lower MAUE then the two TARS in the rest of views, which are 1.067 in Cold start, 0.882 in Heavy, 1.361 in Opinionated and 0.845 in Flexible. The differences of MAE between Basic and Resnick TARS are very small. Furthermore, the high percentages of users' coverage from the three TARS are generated from the Basic TARS in the three views namely Heavy, Opinionated and Flexible users. However, the users' coverage in all and cold start users have been highly achieved

by the TARS with distrust. Similar with MAUE, the users' coverage from Resnick TARS is very similar to Basic TARS in all views.

5. Conclusion

Trust is the measure of enthusiasm to believe in a user based on behavior within a specific context in a period time. In this research, the fundamental aspects and parameters of trust aware recommender systems have been defined. Then, empirical experiments have been conducted to compare the performances of three existing algorithms with different types of trusted users' views. Previous evaluations on the TARS have been conducted in different setting of parameters, which creates a difficulty for researchers to compare the performances of the different techniques. The experiments demonstrated in this research promotes a common way of evaluation to provide performances results that are comparable from different TARS techniques. It was found in this research that the error of accuracy from TARS with distrust element is smaller than the other TARS mainly in all users view. Similarly, all the rating coverage achieved by Distrust TARS in all the users view have been outperform than the two TARS. In term of user's coverage, Distrust TARS has more percentages in all and cold start users and the rests of views have been achieved by Basic TARS.

In future works, different types of parameters should be defined for the performances of TARS. How a different categories of rating values might affect all the performance metrics of TARS is one important question to be answered in the future research. Additionally, this research is not yet considering the types of propagation in TARS between local and global. Among the important issue that need to be studied in future is the local or global propagation in relative to the different groups of users view and ratings. In conclusion, this research has open a lot of performances issues of the existing TARS. Therefore, further research should be conducted to give the reader a visual grasp of the relative benefits of the different techniques

Acknowledgement

We would like to thank Universiti Teknologi MARA for the financial support of this project.

References

[1] Santos Jr EB, Goularte R, Manzato MG (2015), Personalized collaborative filtering: A neighborhood model based on contextual constraints. *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, pp. 919-924.

[2] Mooney RJ, Roy L (2000), Content-based book recommending using learning for text categorization. *Proceedings of the 5th ACM Conference on Digital Libraries*, pp. 195-204.

[3] Mahtar SN, Masrom S, Omar N, Khairudin N, Rahim SK, Rizman ZI (2017), Trust aware recommender system with distrust in different views of trusted users. *Journal of Fundamental and Applied Sciences* 9(5S), *168-182.*

[4] Safoury L, Salah A (2013), Exploiting user demographic attributes for solving cold-start problem in recommender system. *Lecture Notes on Software Engineering* 1(3), 303-307.

[5] Shambour Q, Lu J (2012), A trust-semantic fusion-based recommendation approach for e-business applications. *Decision Support Systems* 54(1), 768-780.

[6] Massa P, Avesani P (2017), Trust-aware recommender systems. *Proceedings of the ACM Conference on Recommender Systems*, pp. 17-24.

[7] Nazemian A, Gholami H, Taghiyareh F (2012), An improved model of trust-aware recommender systems using distrust metric. *Proceedings of the Advances in Social Networks Analysis and Mining*, pp. 1079-1084.

[8] Véras D, Prota T, Bispo A, Prudêncio R, Ferraz C (2015), A literature review of recommender systems in the television domain. *Expert Systems with Applications* 42(22), 9046-9076.

[9] Jain S, Grover A, Thakur PS, Choudhary SK (2015), Trends, problems and solutions of recommender system. *Proceedings of the IEEE International Conference Computing, Communication and Automation*, pp. 955-958.

[10] Cuong KM, Minh NT, Van Canh N (2013), An application of fuzzy geographically clustering for solving the cold-start problem in recommender systems. *Proceedings of the IEEE International Conference Soft Computing and Pattern Recognition*, pp. 44-49.

[11] Devi MK, Samy RT, Kumar SV, Venkatesh P (2010), Probabilistic neural network approach to alleviate sparsity and cold start problems in collaborative recommender systems. *Proceedings of the IEEE International Conference Computational Intelligence and Computing Research*, pp. 1-4.

[12] Ma H, King I, Lyu MR (2007), Effective missing data prediction for collaborative filtering. *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 39-46.

[13] Kim KJ, Ahn H (2008), A recommender system using GA K-means clustering in an online shopping market. *Expert Systems with Applications* 34(2), 1200-1209.

[14] Song W, Liang JZ, Park SC (2014), Fuzzy control GA with a novel hybrid semantic similarity strategy for text clustering. *Information Sciences* 273, 156-170.

[15] Son LH (2014), HU-FCF: A hybrid user-based fuzzy collaborative filtering method in recommender systems. *Expert Systems with Applications* 41(15), 6861-6870.

[16] Rana C, Jain SK (2014), An evolutionary clustering algorithm based on temporal features for dynamic recommender systems. *Swarm and Evolutionary Computation* 14, 21-30.

[17] Dao TH, Jeong SR, Ahn H (2012), A novel recommendation model of location-based advertising: Context-aware collaborative filtering using GA approach. *Expert Systems with Applications* 39(3), 3731-3739.

[18] Gupta S, Nagpal S (2015), Trust aware recommender systems: A survey on implicit trust generation techniques. *International Journal of Computer Science and Information Technologies* 6(4), 3594-3599.

[19] Guo G, Zhang J, Thalmann D (2012), A simple but effective method to incorporate trusted neighbors in recommender systems. *Proceedings of the International Conference on User Modeling, Adaptation, and Personalization*, pp. 114-125.