Improving Recommender System Based on Item's Structural Information in Affinity Network

Muhammad Rifqi Ma'arif
Department of Informatics
State Islamic University Sunan Kalijaga
Yogyakarta, Indonesia
muhammad.rifqi@gmail.com

Agus Mulyanto
Department of Informatics
State Islamic University Sunan Kalijaga
Yogyakarta, Indonesia
agus.mulyanto@uin-suka.ac.id

***Abstract*—This paper proposes a technique to improve the accuracy of recommender system result which employ collaborative filtering technique. The proposed method incorporates structural equivalence score of items in affinity network into collaborative filtering technique. Structural equivalence is one of important concept in social network analysis which captures the similarity of items regarding their structural position on the affinity network. Nowadays, various concepts within social network analysis are widely use in many domains to provide better analytical framework. In this paper, we will use structural equivalence of items to enhance the calculation of items similarity as a part of collaborative filtering method. We tested our approach on Netflix database. Then, based on our results we can conclude that considering the structural information of item in affinity network is indeed beneficial.**

Keywords*—Recommender System, Collaborative Filtering, Structural Equivalence, Social Network Analysis

I. INTRODUCTION

Nowadays, with the development of information technology, the number of accessible information is also increase rapidly. The wide variety of information can be used to support the decision making process but at the same time leads the users into confusion. Recommender system is a popular solution to overcome this information overload problem. Recommender system have been widely used by popular website like Amazon, YouTube, Netflix, etc to improve the user experience that in turns increasing the web traffic and for e-commerce website, possibly increase the sales.

Recommender System (RS) is a software tool or technique that provides interesting and useful suggestions of items to users in a personalized way [1]. Item can be any product, such as movie, song, news, book, and etc. User who utilizes the recommender system will be presented with a set of items which are new, useful and relevant to his personal taste. This "personalized" recommendation distinguishes recommender system from the rest of information retrieval techniques which are commonly found in a search engine. The main benefit of using RS is to help users find the relevant or desired items efficiently and effectively. Provided by thousands of items, it's almost impossible for users to look at the item one by one. Correct recommendation will save them from spending a lot of time for searching over the catalogue.

There are a lot of techniques to generate a recommendation, one of the most popular is collaborative filtering. At glance, collaborative filtering is the process of information filtering based on collaboration pattern among involving agents (e.g. item, people, etc) [2]. For example, on the e-commerce context, the collaboration pattern of a user can be defined based on purchase and browsing behaviors. Collaborative filtering techniques use a data of preferences for items by users (commonly represented by rating) to predict additional topics or products a new user might like [3].

The existing collaborative filtering techniques can be improved in a various way [4]. One of the promising technique to improve recommendation method is comes from Social Network Analysis (SNA). SNA is a analytical tool for measuring the relationship of things or entities (e.g people, group, computer, web page, etc) in terms of network theory which consist of nodes for representing entities and ties for representing relationship between entities [5][6]. On the past few years, many researchers have used SNA to study complex systems in a wide variety of scientific, social and engineering domains. Examples include organization network [7], software development process [8], supply chain management [9], etc.

In the field of recommender system, some researchers also have attempts to put the information generated from a product network to improve the quality and accuracy of recommendation output. Cho & Bang [10] used the concept of centrality in SNA to produce various version of products ranking than recommend top rank products to the customers. Another work done by Liu & Lee [11] who developed a mechanism based on social network information of users to improve nearest neighbor estimation in computing similarity of two items which is the beginning part of collaborative filtering. They have showed that their method provide a better performance in terms of accuracy. Similar kind of work done by Kim and Ahn [12] by utilizing community detection technique in social network analysis to find the cohesive subgroups of users. This cohesive subgroups addressing the group of users which share same interest. They claimed that their study overcome the limitation of collaborative filtering in recognizing the social relation of users which may affect the recommendation results.

In this paper, we address structural equivalence [13] as one of unexplored concept in social network analysis to improve the recommendation resulted by collaborative filtering. Structural equivalence is one of the important concept on the network analysis which measures the similarity of two nodes based on the immediate shared neighbors. Two nodes are considered to be similar if they connected to the same nodes as illustrated in following fig. 1.

We argue that incorporating structural information into collaborative filtering will enhance the recommendation. We came up with this hypothesis by looking at the nature of affinity network. On the affinity network of items, a node represents item while the edge between two node is represents co-purchasing relationship, for example item A is likely to be purchased by user who purchase item B**.** Therefore, inferring items' similarity over the affinity network means we harvest the items which have similar environment in which they bought by customers.

II. Research Method

Our method begin with the construction of affinity network. Afterward, we employ *SimRank* algorithm to infer structural equivalence score of every items and incorporate them into the calculation of item-to-item similarity. We then use the similarity scores of all pair of items to construct user-item similarity matrix. Finally, we simply apply item-based collaborative filtering method to generate the recommendation.

As we stated previously, the affinity network of product is constructed by inspecting the occurrence of co-purchasing of particular products which derived from purchasing data [14]. Hence, the node represents product and the link established between two nodes represents the occurrence of co-purchasing between them, then we normalize co-purchasing frequency with range value between 0 and 1.

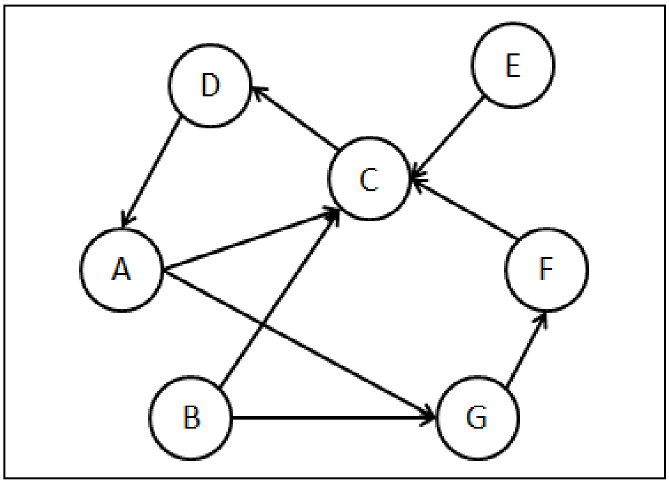

Fig 1. Structural equivalence of nodes. Node **A** and **B** is structurally equivalent as **E** and **F**.

Once we have an affinity network, we use the most widely used algorithm called *SimRank* to calculate the structural equivalence of item within the network of items. *SimRank* is an algorithm to measures the similarity of a pair of entities based on their relationship with another entities beyond them [15]. The intuition behing this algorithm is "two entities are considered to be similar if they have shared relation to the similar entities". The formulation of *SimRank* algorithm is state as folows, first we define a node v in a graph, the neighbor of v are addressed by **Error! Reference source not found.**. Individual neighbors are detoded as **Error! Reference source not found.** for **Error! Reference source not found.**. Then structural equivalence score (*seqv*) between objects **Error! Reference source not found.** and **Error! Reference source not found.** with **Error! Reference source not found.**. If **Error! Reference source not found.** then **Error! Reference source not found.** should be defined as 1. Otherwise, the structural equivalence score can be defined by equation (1). In equation (1), **Error! Reference source not found.** is a constant between 0 and 1. In special condition where **Error! Reference source not found.** or **Error! Reference source not found.**may have no neighbor, then **Error! Reference source not found.** is set to be 0, because there is no way to infer any similarity between **Error! Reference source not found.**and **Error! Reference source not found.**.

$$seqv(v_1, v_2) = \frac{\varphi}{|I(v_1)| \times |I(v_2)|} \sum_{i=1}^{|I(v_1)|} \sum_{j=1}^{|I(v_2)|} s\left(I_i(v_1), I_j(v_2)\right) \quad (1)$$

After inferring structural equivalence score, the next step is to incorporate the score into item-to-item similarity calculation. In collaborative filtering there are a number of methods that can be used to calculate item-to-item similarity. In this paper, we use correlation based similarity by computing pearson product moment correlation or *Pearson-r* correlation for short, which is the most widely used technique in collaborative filtering. *Pearson-r* correlation considers the initial item-to-item similarity. We use **Error! Reference source not found.**to defines the similarity of item **Error! Reference source not found.** and **Error! Reference source not found.** and calculates by following equation (2) [16]. In equation (2) U denotes the user who give a rating of both item **Error! Reference source not found.**and **Error! Reference source not found.**, then **Error! Reference source not found.**represent rating of user u on item **Error! Reference source not found.** Lastly, **Error! Reference source not found.** is the average rating given by users to item **Error! Reference source not found.**.

$$sim(v_1, v_2) = \frac{\sum_{u \in U}(R_{u,v_1} - \overline{R_{v_1}})(R_{u,v_2} - \overline{R_{v_2}})}{\sqrt{\sum_{u \in U}(R_{u,v_1} - \overline{R_{v_1}})^2} \times \sqrt{\sum_{u \in U}(R_{u,v_2} - \overline{R_{v_2}})^2}} \quad (2)$$

In our approach, incorporating structural equivalence into *Pearson-r* correlation is pretty straightforward by taking the weighted average as defined on the following equation (3).In equation (3), **Error! Reference source not found.** is a trade off between *Pearson-r* correlation and structural equivalence. We define **Error! Reference source not found.** as a constant between 0 and 1 for tuning the measures, which one between *Pearson-r* correlation and structural equivalence will have bigger portion to the final result. Afterward, user-item matrix was built based on **Error! Reference source not found.** value. After defining user-item matrix we simply follow the rest of collaborative filtering procedure to generate final recommendation list.

$$\overline{sim}(v_1, v_2) = \alpha \times sim(v_1, v_2) + (1 - \alpha) \times seqv(v_1, v_2) \qquad (3)$$

III. Result and Analysis

For investigating the results of our proposed method we used Netflix movie dataset from www.netflix.com. This dataset contains a rating data from year 1999 to 2005. We then divided data into training set and test set. We employed data from 1999 to 2003 as training set and 2004 and 2005's as test set. Our proposed method was run over the training set and the quality of the results was investigated over the test set. Since the Netflix movie dataset contains information about "what movie borrowed by whom" we can easily create affinity network of a movie from the co-borrowed relationship between movies. Furthermore, this dataset was specifically provided by Netflix for a recommender system competition to improve Netflix's recommender system accuracy. Therefore, this dataset is indeed addressed for recommender system evaluation purpose.

To show the improvement made by our method, we compare the recommendation results of our method to the conventional collaborative filtering method. We use two kind of evaluation: accuracy and serendipity. Accuracy measures how consistent the recommendation result will be taken by the users, whereas serendipity measure how surprising the recommendation results so that it make users interesting. Serendipity gives to the users an interesting shopping experience in terms of surprise and unexpectedness [17].

For evaluating the accuracy, we then used the popular measures named precision [18]. The origin of precision is comes from information retrieval which is defined as the fraction of retrieved documents that are relevant to search. For the recommender system's evaluation purpose, precision can be defined as the ratio of relevant items selected to the number of items which are recommended to the users. Then, we can calculate precision equation (4).

$$Precision = \frac{|\{relevant\ items\} \cap \{recommended\ items\}|}{|\{recommended\ items\}|} \qquad (4)$$

The second evaluation method is serendipity. For calculating serendipity score we follow an approach defined in [14]. The calculation of serendipity start by calculating the unexpectedness score. The unexpectedness score is merely the ratio of items generated by recommender systems (RS) to the number of items generated by primitive models/naïve method (PM) and depicted by equation (5).

$$UNEXP = {}^{RS}/_{PM} \qquad (5)$$

A common example for primitive models is we just recommend the item which is mostly bought by customers, etc. If the unexpected item **Error! Reference source not found.** recommended to the user is relevant to the user preference then we define **Error! Reference source not found.**, and **Error! Reference source not found.** otherwise. Then, serendipity can be defined as following equation (6). In equation (6), N is the total number of elements in *UNEXP*.

$$SRDP = \frac{\sum_{v_1=1}^{N} u(RS_{v_1})}{N} \qquad (6)$$

Finally, table 1 shows the results of our proposed method and the comparison to conservative collaborative filtering as well. From table 1, we can see that the accuracy of conservative collaborative filtering (CF) and our proposed method (collaborative filtering with structural equivalence or CFSE for short) is pretty close each other. Nevertheless, our proposed method shows a slightly better result in terms of serendipity.

TABLE I. THE EVALUATION RESULTS

Evaluation Method	CF	CFSE
Accuracy	0.32891	0.32887
Serendipity	0.00176	0.00297

The evaluation result come up as depicted above in table 1 due to the nature of structural equivalence calculation based on affinity network is close to the basis of pearson correlation calculation. *Pearson-r* correlation calculates items which rate together by same user. Similar things happen to the construction of affinity network which based on the occurrence of co-purchased items, therefore the top results of both calculations will be close each other. Then the our serendipity score raise a better result because unlike *Pearson-r* correlation which simply ignores two items which rated by different users from the calculation, structural equivalence consider the whole networks. Therefore, two items which not directly purchased by same user still have a high correlation score if their neighborhood is similar.

IV. Conclusion

The previous researches in recommender system based on social network analysis were focused to two popular concept: centrality and cluster/community. Centrality concept was used to determined *n*-top items in various context, while cluster/community was used as pre-processed steps before going into main process of recommendation. Another utilization of community is to enrich the recommender system calculation by considering user preference which not accessible by conventional recommender system approach like collaborative filtering.

This paper have introduced structural equivalence as one of the important concept in social network analysis which never been explored before in social network based recommender systems. We use structural equivalence concept to enhance the calculation of item-to-item correlation in item-based collaborative filtering. Practically, we take the normalized mean of *Pearson-r* correlation and structural equivalence score. We have showed that considering structural equivalence will affect the serendipity of recommendation. That means, the recommendation results will suggest items that never been though before but relevant to the users/customers.

REFERENCES

[1] B. Robin, "Hybrid recommender systems: Survey and Experiments," User Modeling and User-Adapted Interaction, vol 12, no. 4, pp . 331-370, 2002.

[2] 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.

[3] X. Su and T. M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques." Advances in artificial intelligence, pp. 4, 2009.

[4] G. Adomavicius and A. Tuzhilin, "Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions," IEEE Transactions on Knowledge and Data Engineering vol 17, no. 6, pp. 734-749, 2005.

[5] C.A.R. Pinheiro, “Social Network Analysis in Telecommunications,” John Wiley & Sons, ISBN. 978-1-118-01094-5, pp. 4, 2011.

[6] D. Alessia, "An Overview of Methods for Virtual Social Network Analysis,", In A. Ajith. Computational Social Network Analysis: Trends, Tools and Research Advances. Springer. p. 8. ISBN 978-1-84882-228-3, 2011.

[7] T. Murshed, J. Davis, and L. Hossain, "Social Network Analysis and Organizational Disintegration: The Case of Enron Corporation," In International Conference on Information Systems (ICIS), 2007.

[8] W. J. Goh and K. Y. Tang, "Investing Success of Open Source Software Projects: A Social Network Perspective," In Proceeding of International Conference on Information Systems (ICIS), 2007.

[9] S. P. Borgatti and X. Li, "On Social Network Analysis in A Supply Chain Context," Journal of Supply Chain Management, vol. 45, no. 2 pp. 5-22, 2009

[10] Y. Cho and J. Bang, “Social Network Analysis for New Product Recommendation,” Journal of Intelligence and Information Systems, vol. 15, no. 4, pp. 177-193, 2009.

[11] 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.

[12] J. Y. Kim and H. C. Ahn, "Hybrid Recommender Systems using Social Network Analysis," In Proceedings of World Academy of Science, Engineering and Technology, no. 64, 2012.

[13] F. Lorrain and H. C. White, "Structural Equivalence of Individuals in Social Networks," The Journal of Mathematical Sociology, vol. 1, no. 1, pp. 49-80, 1971

[14] H.K. Kim, J. K. Kim and Y.C. Qiu, "A Product Network Analysis for Extending the Market Basket Analysis," Expert Systems with Applications, vol. 39, no. 8, pp. 7403-7410, 2012.

[15] G. Jeh and J. Widom, "SimRank: A Measure of Structural-Context Similarity." In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 538-543, 2002.

[16] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-Based Collaborative Filtering Recommendation Algorithms." In Proceedings of the 10th International Conference on World Wide Web, pp. 285-295, 2001.

[17] M. Ge, C. D. Battenfeld, and D. Jannach, "Beyond Accuracy: Evaluating Recommender Systems by Coverage and Serendipity," In Proceedings of the Fourth ACM Conference on Recommender Systems, pp. 257-260, 2010.

[18] J. Makhoul, F. Kubala, R. Schwartz, and R. Weischedel, "Performance Measures for Information Extraction," In Proceedings of DARPA Broadcast News Workshop, pp. 249-252, 1999.