International Journal of Engineering & Technology, 7 (4.40) (2018) 193-197

International Journal of Engineering & Technology

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

Research paper

WP-Rank: Rank Aggregation based Collaborative Filtering Method in Recommender System

Sri Lestari [1,2]*, Teguh Bharata Adji[1], Adhistya Erna Permanasari [1]

[1]*Department of Electrical Engineering and Information Technology, Universitas Gadjah Mada, Yogyakarta, Indonesia.*
[2]*Faculty of Computer Science, Institut Informatika dan Bisnis Darmajaya, Bandar Lampung, Indonesia.*
**Corresponding author E-mail: srilestari.s3te15@mail.ugm.ac.id*

Abstract

Collaborative filtering with a traditional rating-based approach uses interaction records between users and systems to measure similarity, prediction, and generate recommendations. However, traditional rating based cannot capture user preferences of different products. To overcome this issue, the ranking based approach, such as the Borda method has been used. The method takes advantage of rating data to determine the position of the product in the list of preferences as the basis for determining product points. However, the list of the preferences which is merely based on rating data, results in an insufficient accuracy of the recommendation. This paper, therefore, proposed a novel approach namely the WP-Rank aggregation method which maximizes the use of rating data to generate product weight. The experimental results show that the WP-Rank method was superior to the Borda method with NDCG average value difference of 0.0220. However, the WP-Rank method required longer running time with 0.0206 seconds lag from the Borda method.

Keywords: *Recommendation system, Collaborative filtering, WP-Rank, Borda*

1. Introduction

The development of technology has provided convenience for the users by providing various information either in the forms of images, video, blocks, multimedia and so forth. However, the diverse information creates challenges for analysts to extract user-beneficial knowledge and capture meaningful events from a massive amount of data. Many studies have been conducted to solve these challenges such as the development of recommendation systems. Recommendation systems are systems which can simplify, support, and ease processes from very large inputs of data and information to produce efficient output and specify the needs of users or organizations[1].

Recommendation systems are rapidly evolving, which are used in various domains such as books, movies, songs, restaurants, news and other media [2]. In addition, various approaches have been developed to improve the performance of recommendation systems, such as content-based, collaborative filtering, demographic, knowledge-based, community-based, and hybrid systems[3]. One of the popular approaches is collaborative filtering [4], which is an algorithm based on the prediction and recommendation of other users' judgments or behaviors with similar preference in the past. Several categories of the implementation of collaborative filtering are memory-based, model-based, and hybrid [1][5][6].

The memory-based approach uses data ratings derived from user interaction records with the system, as a basis for measuring the similarity, prediction, and generation of recommendation. The examples of the memory-based approach are Pearson correlation [7][8][9], k-Nearest Neighbors (k-NN) [10], and Cosin methods [11] for rating based. Meanwhile, the examples of rankings based are VSRank [12] and Borda methods [13]. However, the memory-based approach faces the main problems of sparsity and scalability of data. Data sparsity arises because most users only rate a small number of products; thus, the user-item matrix becomes rare and affects the recommendation quality. Meanwhile, computing scalability in the memory-based method is often decreased when the number of users and items are increased [5].

In contrast to the memory-based approach, the model-based approach builds user profiles based on user preferences of some products, then calculates the probability of users who share the same personality with others. The examples of model-based approach are Bayesian Network algorithm [5][13], clustering algorithm [5][14][15][16], and Singular Value Decomposition (SVD) algorithm [5][11][14]. Nevertheless, the model-based approach often takes a long time to be built and updated, and it cannot cover a wide range of various users as the memory-based approach.

Moreover, there is another approach namely the hybrid approach, which consists of several applications that combine the techniques of memory-based and model-based collaborative filtering [1], such as content-boosted CF algorithm and Personality Diagnosis (PD). The advantages of this approach are it can improve prediction performance and address data sparsity problems. However, hybrid approaches generate complexity of the system due to the variety of techniques used.

The various advantages possessed by each approach are utilized to improve the performance of the recommendation system. One of the highlights is ranking-based that can generate a recommendation list without predicting the rating, and it is able to operate well despite the data sparsity conditions. Various methods of ranking-based collaborative filtering have been developed including aggregation rankings, but studies using such approaches are rare [13]. Some studies that use aggregation rankings are Das et al. [17] which combined clustering techniques with voting algorithms. The study used DBSCAN as the clustering technique by classifying users to identify groups of users with similar characteristics. Meanwhile, the Borda method is used as the voting algorithm to combine the opinions of some users. The combination of

DBSCAN and Borda generated the recommendation of the most favored films according to the genre. However, because genre selection was done randomly, this resulted in a decrease of the recommendation quality. Furthermore, Wu et al. [18], used user-based collaborative filtering (CF), graph-based method and social-based CF methods, which were integrated using aggregation methods of CombSum, CombMED, and Borda to form hybrid recommendations. The approach improved the performance of recommendations but the Borda method accuracy was still low.

The Borda method is a method used in group decision making for the selection of winners. Determination of winners is based on the acquisition of points obtained from the position of the product in the users' list of preferences. Since the products rank in the list of preferences is only based on rating data, then the recommendation result becomes less accurate. The problem with the Borda method is the inaccuracy of the recommendation result as it is only based on the use of rating data for product positioning in the preference list. Therefore, this study aimed to propose a new method of Weight Point Rank (WP-Rank). The method works by optimizing the use of rating data as a weighting at the time of product point determination; thus, increasing the relevance of the recommendations.

2. WP-Rank

This paper proposed the Weight Point Rank (WP-Rank) method, which aims to increase the relevance of the recommendations of product to users. The WP-Rank method works by maximizing the use of rating data to generate product weight.

Suppose that there is a shop selling and renting CD/DVD movies. Customers who buy or rent CD/DVD movies are required to provide ratings. For example, there are 5 users (u_1, u_2, …., u_5) who rate 10 movies (p_1, p_2, ..., p_{10}) as in Table 1. The rating values range from 1 to 5. The greater the rating indicates the higher the user's interest in a product and otherwise.

Table 1: User Item Matrix

	p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}
u_1	5	2	3	4	5	5	4	3	2	1
u_2	0	3	5	3	2	3	4	5	3	4
u_3	2	4	5	4	3	4	2	3	4	5
u_4	0	3	4	2	4	1	3	4	3	3
u_5	5	4	2	4	3	5	5	3	4	4

Based on the data in Table 1 then we can provide the users with recommendations by using the developed WP-Rank method. The WP-Rank process consists of 4 stages, namely:

1) Calculate the same rating amount

$$S_{(u_g,p_h)} = \sum_{k=1}^{n} SR(R_{(u_g,p_h)}, R_{(k,p_h)}) \quad (1)$$

$$SR(R_{(u_g,p_h)}, R_{(k,p_h)}) = \begin{cases} 1, if\ (R_{(u_g,p_h)}, R_{(k,p_h)}) \\ 0, if\ (R_{(u_g,p_h)} = 0) \\ 0, otherwise \end{cases} \quad (2)$$

User is represented as U = {u_1, u_2, ..., u_g, …, u_{l-1},u_l}, the product is represented as P = {p_1, p_2, …, p_h, …, p_{m-1},p_m}, rating from user u_g for product p_h is represented as $R_{(ug,ph)}$ and rating from user u_k for product p_h is represented as $R_{(k,ph)}$. Equation 1 shows the same rating amount ($S_{(ug,ph)}$) gained by calculating the same rating amount in every product ($SR_{(ug,ph)}$,$R_{(k,ph)}$). The same rating is assigned with the value of 1, and the others are 0, as in Equation (2).

2) Determining product point

$$P_{(u_g,p_h)} = 1 + \sum_{k=1}^{m} PR(u_g, p_h, k) \quad (3)$$

$$PR_{(u_g,p_h,k)} = \begin{cases} 1, if\ R_{(u_g,p_h)} > R_{(u_g,k)}, \\ 1, if\ R_{(u_g,p_h)} = R_{(u_g,k)}, S_{(u_g,p_h)} > S_{(u_g,k)} \\ 1, if\ R_{(u_g,p_h)} = R_{(u_g,k)}, S_{(u_g,p_h)} = S_{(u_g,k)}, u_g < k, \\ 0, otherwise \end{cases} \quad (4)$$

Equation 3 is used to determine the ranked product point *($P_{(ug,ph)}$)* obtained by adding the value of 1 to the result of the sum of point ranks *$PR_{(ug,ph,k)}$* as in Equation 4. The product point is given the value of 1 if:

a. the product rating to-P_h ($R_{(ug, ph)}$) is greater than the product rating to-k ($R_{(ug,k)}$).
b. the product rating to-P_h($R_{(ug, ph)}$) is same with the product rating to-k ($R_{(ug,k)}$), and the same product rating amount to-P_h($S_{(ug,ph)}$) is greater than the same product rating amount to-k ($S_{(ug,k)}$).
c. the product rating to-P_h($R_{(ug, ph)}$) is same with the product rating to-k ($R_{(ug,k)}$), and the same product rating amount to-P_h ($S_{(ug,ph)}$) is same with the same product rating amount to-k ($S_{(ug,k)}$), and the product index to-P_h is smaller than the product index to-k.
d. Apart from these three conditions then the point will be 0.

3) Calculating Weight Point

$$WP_{(u_g,p_h)} = (S_{(u_g,p_h)} + R_{(u_g,p_h)}) P_{(u_g,p_h)} \quad (5)$$

Equation 5 is used to calculate the weight point (*$WP_{(ug, ph)}$*), obtained from the multiplication process of the product point *$P_{(ug,ph)}$*) with the sum of the same rating amount *($S_{(ug,ph)}$)* and rating *($R_{(ug,ph)}$)*.

4) Calculating Weight Point Rank (WP-Rank)

$$WP-Rank_{ph} = \sum_{k=1}^{n} WP_{(k,ph)}$$
(6)

Equation 6 states that the Weight Point Rank (WP-Rank$_{(ph)}$) is obtained by summing the weight point of each product ($WP_{(k,ph)}$). The output of WP-Rank is a score for each product, which will then be sorted from the highest score to the lowest score and taken Top-K product.

3. Research Method

The stages of this research included: collecting datasets from MovieLens [19]–[22], which was then clustered using the K-means algorithm into 7 clusters. Each cluster formed was then aggregated using the Borda method and the WP-Rank method. The results were then evaluated using NDCG to see the relevance of the recommendations. In addition, the evaluation of running time on the Borda method and the WP-Rank method was also done to see the complexity of both methods. These stages can be seen in Figure 1 and each of the stages is described in the following sections.

3.1. Dataset

This dataset can be accessed at http://grouplens.org/datasets/movielens/100k/. The dataset consists of 100,000 ratings with scales 1-5 obtained from 943 users and 1682 movies. The dataset contains sparsity of approximately 93.7% and has general characteristics of demographic information (user id, age, gender, occupation, and zip code), and consists of 19 genres.

3.2. Clustering

This study used demographic and genre information in the dataset for the clustering process. The genre clustering process was done by seeing that one genre consists of many movies, and one movie can be a member of several genres. Based on these conditions, then the movies were grouped into their respective genres. There are 19 genres in the dataset, but only favorite genres were used in this study [23]. Among 19 genres, only the top 10 favorite genres were selected based on the highest number of ratings that the genre has. The list of genres can be seen in Figure 2.

The next process was clustering the user by utilizing demographic information. A total of 943 users in the dataset were grouped into 7 clusters based on the user's age using the K-means algorithm. K-means is a clustering algorithm that is widely used by researchers because it is a simple algorithm and the process is fast [24]. The K-means algorithm divides the number of objects in the dataset into partitions based on certain categories by looking at the given centroid. The distance between an object and the closest centroid is used to classify the object. Hence, the object becomes a member of the category that is formed. Euclidean Distance Space is one equation for calculating the distance by knowing the shortest distance between two points. The distance is calculated using Eq. (7).

$$D(x_i, c_j) = \sqrt{\sum_{i=1}^{N} (x_i - c_j)^2} \quad (7)$$

The K-means algorithm is started by determining the number of clusters (k) and followed by determining the centroid (c_j) of each cluster randomly. The next step is to calculate the distance of each object x_i to c_j denoted by $D(x_i, c_j)$. Based on the process, the object will be grouped into the nearest centroid. The next step is to update the centroid of each cluster by calculating the average distance of all objects to the centroid. The process will be repeated until it reaches convergence.

3.3. Aggregation Process

The aggregation process was done on every cluster using the WP-Rank method and the Borda method. Both methods aim to unify the opinions of some users to produce a product ranking. The product ranking generated by the WP-Rank method is obtained through several stages of the calculation process, which includes calculating the same rating amount, determining product points, calculating weight points, and calculating weight point ranks as described in Section 2. In the meantime, the Borda method calculates the points of each product. Position in the preference list becomes the basis for determining product points. The point of 1 is given to the product in the last position while the point of 2 is given to the product with one position before the last, and so on. Mathematically, it can be said that the first place position gets N point, where N is the number of products. The second place receives (N - 1) point, and so on. Based on that points, the product with the biggest point or value is declared as the winner [17].

3.4. Evaluation

This research used two evaluation approaches i.e. running time evaluation and rank evaluation. Running time evaluation was done by calculating the average processing time required by the Borda and WP-Rank methods. In the meantime, ranking evaluation was done using the standard commonly used in the recommendation system, namely Normalized Discounted Cumulative Gain (NDCG) [12][20][25][26]. The evaluation takes into account the higher product rankings in the ranking list[27]. The NDCG definition for Top-K product ranking can be seen in Equations (8) or (9):

$$NDCG = z_n \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{\log_2 (i+1)} \quad (8)$$

or

$$NDCG_p = \frac{DCG_p}{IDCG_p} \quad (9)$$

Equation (8) is the Discounted Cumulative Gain (DCG), which is obtained by summing the result of $\frac{2^{rel_i}-1}{log_2(i+1)}$. rel_i is the value of the relevance (rating) given by the user at that rank and p is the number of data (product) to be evaluated. The next process is to calculate the NDCG (Equation (9)) obtained from $\frac{DCG_p}{IDCG_p}$, where Ideal Discounted Cumulative Gain (IDCG) is the data that has been sorted by rating value to reach the ideal value.

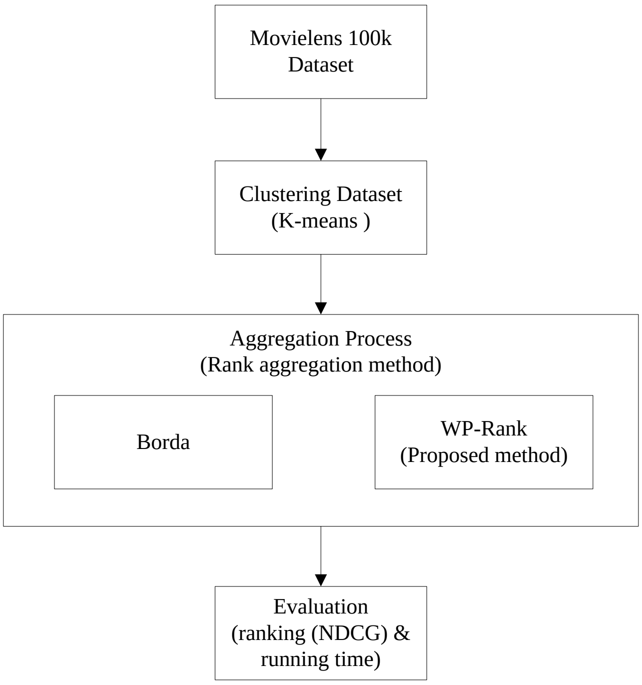

Fig. 1: Flowchart of Rank Aggregation Based on Collaborative Filtering

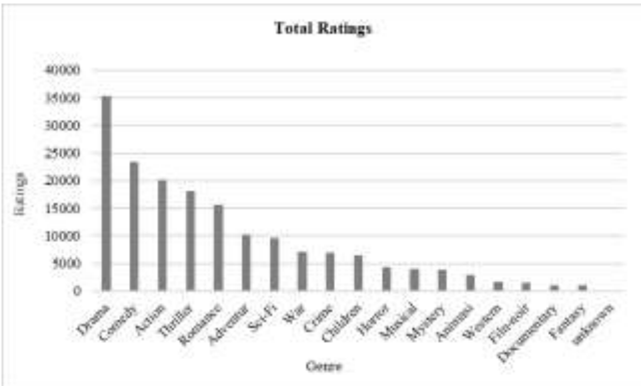

Fig. 2: The user's interest in the movie genre

4. Results and Discussion

Experimental results show that the WP-Rank method was superior to the Borda method. The NDCG evaluation results are shown in Table 2-8. Table 2-5, 7 and 8 show that the mean values of NDCG@1, NDCG@3, NDCG@5, NDCG@10 of the WP-Rank method were higher than those of the Borda method. However, the WP-Rank method was only inferior to the Borda method (Table 6) in NDCG@3 and NDCG@5. Overall, the respective mean values of NDCG@1, NDCG@3, NDCG@5, NDCG@10 are 0.5102, 0.5269, 0.5803, and 0.7411 for the WP-Rank method, and 0.485, 0.5021, 0.5630, and 0.7201 for the Borda method. Thus, the WP-Rank method was superior to the Borda method with NDCG average value difference of 0.0220.

Table 2: NDCG in Cluster 1

Genre	NDCG@1		NDCG@3		NDCG@5		NDCG@10	
	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank
Drama	0.5272	0.3834	0.4230	0.4092	0.4778	0.4725	0.6785	0.6472
Comedy	0.3306	0.3404	0.4562	0.4154	0.5371	0.5103	0.6599	0.6557
Action	0.1593	0.4168	0.3414	0.5215	0.4109	0.5683	0.5957	0.7031
Thriller	0.3235	0.5502	0.3496	0.4893	0.4506	0.5520	0.6089	0.7313
Romance	0.0860	0.4154	0.4576	0.5014	0.5168	0.5496	0.5983	0.6820
Adventur	0.6406	0.5668	0.6649	0.5973	0.6973	0.6602	0.8069	0.7947
Sci-Fi	0.5677	0.6194	0.6591	0.6615	0.7334	0.7277	0.8195	0.8392
War	0.5565	0.6210	0.5453	0.6338	0.5972	0.6519	0.7792	0.8184
Crime	0.3456	0.2765	0.3988	0.4422	0.5393	0.5402	0.6733	0.6913
Children	0.5376	0.4165	0.5573	0.5382	0.6088	0.5992	0.7904	0.7579
Average	0.4075	**0.4606**	0.4853	**0.5210**	0.5569	**0.5832**	0.7011	**0.7321**

Table 3: NDCG in Cluster 2

Genre	NDCG@1		NDCG@3		NDCG@5		NDCG@10	
	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank
Drama	0.3015	0.3837	0.3550	0.4100	0.4311	0.4775	0.6324	0.6637
Comedy	0.4465	0.4273	0.4801	0.4297	0.5316	0.4943	0.6960	0.6797
Action	0.4731	0.4785	0.4934	0.4825	0.5456	0.5266	0.7066	0.7073
Thriller	0.4050	0.4144	0.3906	0.4735	0.4305	0.5447	0.6556	0.7135
Romance	0.5535	0.5375	0.4953	0.5288	0.5681	0.5843	0.7205	0.7402
Adventur	0.5789	0.5690	0.4618	0.5808	0.4991	0.6168	0.7185	0.7736
Sci-Fi	0.5828	0.5820	0.5259	0.5984	0.5790	0.6363	0.7625	0.7857
War	0.5831	0.5968	0.5763	0.6055	0.6090	0.6279	0.7708	0.7886
Crime	0.4785	0.5446	0.5483	0.5674	0.5946	0.6056	0.7463	0.7672
Children	0.4734	0.4702	0.4710	0.5311	0.5613	0.5881	0.7236	0.7493
Average	0.4876	**0.5004**	0.4798	**0.5208**	0.5350	**0.5702**	0.7133	**0.7369**

Table 4: NDCG in Cluster 3

Genre	NDCG@1		NDCG@3		NDCG@5		NDCG@10	
	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank
Drama	0.3969	0.3513	0.4192	0.4126	0.5047	0.4900	0.6704	0.6672
Comedy	0.4772	0.4408	0.5241	0.4615	0.5860	0.5309	0.7251	0.7018
Action	0.5329	0.5031	0.5270	0.4624	0.6003	0.5189	0.7333	0.7029
Thriller	0.2646	0.3889	0.3009	0.4350	0.3698	0.5060	0.5947	0.6858
Romance	0.5768	0.5321	0.5511	0.4956	0.6413	0.5627	0.7546	0.7314
Adventur	0.5669	0.5567	0.4465	0.5461	0.5097	0.6181	0.7150	0.7674
Sci-Fi	0.5925	0.5829	0.5873	0.6034	0.6212	0.6034	0.7806	0.7723
War	0.5752	0.5570	0.5655	0.5393	0.6204	0.5874	0.7776	0.7653
Crime	0.2478	0.3607	0.3192	0.4516	0.4353	0.5300	0.5831	0.6634
Children	0.5083	0.4935	0.6052	0.5528	0.6205	0.6206	0.7388	0.7354
Average	0.4739	**0.4767**	0.4846	**0.4960**	0.5509	**0.5568**	0.7073	**0.7193**

Table 5: NDCG in Cluster 4

Genre	NDCG@1		NDCG@3		NDCG@5		NDCG@10	
	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank
Drama	0.3416	0.3488	0.3836	0.3941	0.4588	0.4699	0.6391	0.6538
Comedy	0.4961	0.4872	0.4955	0.5188	0.6014	0.5543	0.7173	0.7240
Action	0.4493	0.4630	0.4511	0.4878	0.5142	0.5578	0.6864	0.7184
Thriller	0.2151	0.4491	0.3753	0.4784	0.4305	0.5372	0.6182	0.6981
Romance	0.4628	0.4446	0.5363	0.4676	0.5799	0.5361	0.7125	0.7016
Adventur	0.5003	0.5140	0.4454	0.5374	0.5095	0.5857	0.7014	0.7450
Sci-Fi	0.5497	0.5507	0.4801	0.5143	0.5501	0.5533	0.7203	0.7437
War	0.5699	0.5692	0.5657	0.5630	0.6356	0.6076	0.7735	0.7780
Crime	0.3732	0.5020	0.4543	0.5727	0.5254	0.6273	0.6940	0.7622
Children	0.4680	0.4999	0.4898	0.5280	0.5543	0.5876	0.7219	0.7519
Average	0.4426	**0.4829**	0.4677	**0.5062**	0.5360	**0.5617**	0.6985	**0.7277**

Table 6: NDCG in Cluster 5

Genre	NDCG@1		NDCG@3		NDCG@5		NDCG@10	
	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank
Drama	0.3590	0.3236	0.3969	0.3961	0.4869	0.4654	0.6537	0.6534
Comedy	0.4252	0.3623	0.4722	0.4264	0.5802	0.4924	0.6896	0.6696
Action	0.5802	0.5359	0.5096	0.5051	0.5723	0.5491	0.7304	0.7220
Thriller	0.2556	0.5251	0.3624	0.5074	0.4257	0.5585	0.6105	0.7169
Romance	0.6259	0.6106	0.6428	0.5946	0.6921	0.6600	0.7925	0.7969
Adventur	0.6966	0.6923	0.6716	0.5607	0.6778	0.6152	0.8121	0.7910
Sci-Fi	0.6438	0.6404	0.6889	0.6394	0.6948	0.6614	0.8232	0.8174
War	0.6533	0.5611	0.6321	0.5784	0.6341	0.5961	0.7982	0.7773
Crime	0.5161	0.6406	0.6194	0.6079	0.6114	0.6409	0.7696	0.7943
Children	0.3226	0.4076	0.4170	0.3761	0.5815	0.4469	0.6667	0.6694
Average	0.5078	**0.5300**	0.5413	**0.5192**	0.5957	**0.5686**	0.7347	**0.7408**

Table 7: NDCG in Cluster 6

Genre	NDCG@1		NDCG@3		NDCG@5		NDCG@10	
	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank
Drama	0.3349	0.3361	0.4072	0.3988	0.4728	0.4819	0.6383	0.6651
Comedy	0.4150	0.3880	0.4896	0.4654	0.5547	0.5316	0.7032	0.6915
Action	0.6315	0.6064	0.5633	0.5953	0.6521	0.6287	0.7818	0.7746
Thriller	0.2430	0.3397	0.3365	0.4294	0.3948	0.5051	0.6019	0.6661
Romance	0.6667	0.6147	0.5821	0.5705	0.6185	0.6044	0.7765	0.7682
Adventur	0.6897	0.6897	0.5609	0.6202	0.5863	0.6413	0.7703	0.7904
Sci-Fi	0.6671	0.6575	0.5600	0.6020	0.6190	0.6310	0.7779	0.7902
War	0.6592	0.6412	0.5595	0.5745	0.5831	0.6159	0.7743	0.7898
Crime	0.3710	0.5274	0.4972	0.5715	0.6289	0.6242	0.7417	0.7790
Children	0.3683	0.5403	0.5066	0.5656	0.5747	0.6122	0.7036	0.7493
Average	0.5046	**0.5341**	0.5063	**0.5393**	0.5685	**0.5876**	0.7270	**0.7464**

Table 8: NDCG in Cluster 7

Genre	NDCG@1		NDCG@3		NDCG@5		NDCG@10	
	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank	Borda	WP-Rank
Drama	0.3350	0.4394	0.3943	0.4628	0.4547	0.5403	0.6257	0.6985
Comedy	0.5097	0.4240	0.5916	0.4437	0.6482	0.5175	0.7601	0.6964
Action	0.4459	0.5081	0.3876	0.4999	0.5674	0.5676	0.6861	0.7442
Thriller	0.5652	0.4692	0.5041	0.4719	0.5696	0.5611	0.7067	0.6979
Romance	0.4792	0.5231	0.4647	0.5614	0.4784	0.6207	0.7104	0.7785
Adventur	0.6232	0.4129	0.6413	0.5273	0.6719	0.6045	0.8215	0.7634
Sci-Fi	0.9263	0.9263	0.7913	0.7740	0.7696	0.7778	0.9017	0.9075
War	0.4419	0.6097	0.5236	0.5880	0.5203	0.6444	0.7537	0.8106
Crime	0.8105	0.8105	0.6693	0.8020	0.7172	0.7776	0.8467	0.8845
Children	0.5806	0.7473	0.5314	0.7252	0.5828	0.7288	0.7742	0.8617
Average	0.5718	**0.5871**	0.5499	**0.5856**	0.5980	**0.6340**	0.7587	**0.7843**

In addition to evaluation with NDCG, this study also performed the running time analysis of the WP-Rank and Borda methods, which were 0.0264 seconds and 0.0058 seconds, respectively (Figure 3). Hence, the WP-Rank method needed a longer running time with 0.0206 seconds lag from the Borda method.

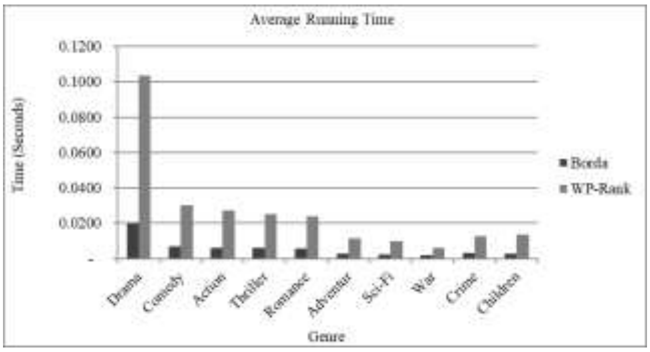

Fig. 3: Average Running Time

5. Conclusion

This paper proposed a new Collaborative filtering framework by combining clustering and ranking approaches. The clustering process used the K-means algorithm by using user demographic information and genre information available in the MovieLens dataset. The clustered dataset was then aggregated to generate the product ranking. The aggregation method used the Borda method. The method used the rating data to determine the position of the product in the list of preferences to determine product points. The list of preferences merely based on rank data tends to result in less accurate recommendations. Therefore, the WP-Rank aggregation method was proposed to maximize the use of rating data to generate product weight. The experimental results show that the WP-Rank method was superior to the Borda method with the NDCG average value difference of 0.0220. However, the WP-Rank required longer running time than the Borda method with 0.0206 seconds lag. For future work, a combination with other methods to improve the accuracy of the ranking process and testing with different domains are strongly suggested.

Acknowledgment

This study was supported by the Doctoral Grant Research under Ministry of Research, Technology and Higher Education of the Republic of Indonesia No. SP DIPA-042.06.1.401516/2018. Workshop on Improving Research Quality of Research Scheme of Doctoral Dissertation in 2018.

References

[1] S. Sehgal, S. Chaudhry, P. Biswas, and S. Jain, “A New Genre Of Recommender Systems Based On Modern Paradigms Of Data Filtering,” *Procedia - Procedia Comput. Sci.*, vol. 92, pp. 562–567, 2016.

[2] L. Hoang, “Dealing with the new user cold-start problem in recommender systems : A comparative review,” *Inf. Syst.*, vol. 58, pp. 87–104, 2016.

[3] F. Ricci, L. Rokach, and B. Shapira, *Introduction to Recommender Systems Handbook*. Springer Science+Business Media, 2011.

[4] M. Nilashi, D. Jannach, O. Bin Ibrahim, and N. Ithnin, “Clustering- and regression-based multi-criteria collaborative filtering with incremental updates,” *Inf. Sci. (Ny).*, vol. 293, pp. 235–250, 2015.

[5] G.-R. Xue *et al.*, “Scalable collaborative filtering using cluster-based smoothing,” *Proc. 28th Annu. Int. ACM SIGIR Conf. Res. Dev. Inf. Retr. - SIGIR ’05*, p. 114, 2005.

[6] J. Wang and L. Ke, “Feature subspace transfer for collaborative filtering,” *Neurocomputing*, vol. 136, pp. 1–6, 2014.

[7] L. Hu, G. Song, Z. Xie, and K. Zhao, “Personalized Recommendation Algorithm Based on Preference Features,” *Tsinghua Sci. Technol.*, vol. 19, no. 3, pp. 293–299, 2014.

[8] N. Polatidis and C. K. Georgiadis, “A multi-level collaborative filtering method that improves recommendations,” *Expert Syst. Appl.*, vol. 48, pp. 100–110, 2016.

[9] C. Veena and B. V. Babu, “A User- Based Recommendation with a Scalable Machine Learning Tool,” *Int. J. Electr. Comput. Eng.*, vol. 5, no. 5, pp. 1153–1157, 2015.

[10] S. K. Tiwari and H. Potter, “An Approach for Recommender System by Combining Collaborative Filtering with User Demographics and Items Genres,” *Int. J. Comput. Appl.*, vol. 128, no. 13, pp. 16–24, 2015.

[11] J. Wu, L. Yang, and Z. Li, “Variable weighted BSVD-based privacy-preserving collaborative filtering,” *Proc. - 2015 10th Int. Conf. Intell. Syst. Knowl. Eng. ISKE 2015*, pp. 144–148, 2016.

[12] S. Wang, J. Sun, and B. J. Gao, “VSRank : A Novel Framework for Ranking-Based,” *ACM Trans. Intell. Syst. Technol.*, vol. 5, no. 3, 2014.

[13] Y. Tang and Q. Tong, “BordaRank: A ranking aggregation based approach to collaborative filtering,” *2016 IEEE/ACIS 15th Int. Conf. Comput. Inf. Sci. ICIS 2016 - Proc.*, 2016.

[14] K. A. Almohsen and H. Al-jobori, “Recommender systems in light of big data,” *Int. J. Electr. Comput. Eng.*, vol. 5, no. 6, pp. 1553–1563, 2015.

[15] Y. Cai, H. F. Leung, Q. Li, H. Min, J. Tang, and J. Li, “Typicality-based collaborative filtering recommendation,” *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 3, pp. 766–779, 2014.

[16] J. Chen, H. Wang, and Z. Yan, “Evolutionary heterogeneous clustering for rating prediction based on user collaborative fi ltering ☆,” *Swarm Evol. Comput.*, vol. 38, no. April 2017, pp. 35–41, 2018.

[17] J. Das, P. Mukherjee, S. Majumder, and P. Gupta, “Clustering-based recommender system using principles of voting theory,” *Proc. 2014 Int. Conf. Contemp. Comput. Informatics, IC3I 2014*, pp. 230–235, 2014.

[18] H. Wu, Y. Hua, B. Li, and Y. Pei, “Personalized Recommendation via Rank Aggregation in Social Tagging Systems,” *Proc. 10th Int. Conf. Fuzzy Syst. Knowl. Discov.*, no. 2010, pp. 888–892, 2013.

[19] B. H. Huang and B. R. Dai, “A Weighted Distance Similarity Model to Improve the Accuracy of Collaborative Recommender System,” *Proc. - IEEE Int. Conf. Mob. Data Manag.*, vol. 2, pp. 104–109, 2015.

[20] B. Shams and S. Haratizadeh, “Graph-based collaborative ranking,” *Expert Syst. Appl.*, vol. 67, pp. 59–70, 2017.

[21] Y. Hu, Y. Yang, C. Li, Y. Wang, and L. Li, “A hybrid genre-based personalized recommendation algorithm,” *2016 IEEE 11th Conf. Ind. Electron. Appl.*, pp. 1369–1373, 2016.

[22] F. M. Harper and J. A. Konstan, “The MovieLens Datasets : History and Context,” *ACM Trans. Interact. Intell. Syst*, vol. V, 2015.

[23] W. Insuwan, U. Suksawatchon, and J. Suksawatchon, “Improving missing values imputation in collaborative filtering with user-preference genre and singular value decomposition,” *Knowl. Smart Technol. (KST), 2014 6th Int. Conf.*, pp. 87–92, 2014.

[24] Z. Min and Y. Shuzhen, “A Collaborative Filtering Recommender Algorithm Based On the User Interest Model,” *IEEE 17th Int. Conf. Comput. Sci. Eng.*, pp. 3–7, 2014.

[25] W. Yao, J. He, G. Huang, and Y. Zhang, “SoRank: incorporating social information into learning to rank models for recommendation,” *Proc. 23rd Int.*, pp. 409–410, 2014.

[26] Y. Guo, X. Wang, and C. Xu, “CroRank: Cross Domain Personalized Transfer Ranking for Collaborative Filtering,” *Proc. - 15th IEEE Int. Conf. Data Min. Work. ICDMW 2015*, pp. 1204–1212, 2016.

[27] Y. Shi, M. Larson, and A. Hanjalic, “Unifying rating-oriented and ranking-oriented collaborative filtering for improved recommendation,” *Inf. Sci. (Ny).*, vol. 229, pp. 29–39, 2013.