

# Evaluation of Group Modelling Strategy in Model-Based Collaborative Filtering Recommendation

R. Mat Nawi, S. A. Mohd Noah, and L. Q. Zakaria

**Abstract**—Recommender systems for groups are becoming increasingly popular since many information needs instigate from group and social activities, such as listening to music, watching movies, and traveling. One of the important aspects in group recommendation is group modelling aggregation strategy which is a process to generate the overall ratings of the group. Such ratings are considered as representations of the groups. There are few group aggregations approaches. In this paper we evaluated two group aggregation approaches which are the Most Pleasure and Average strategy group modelling. We implemented both approaches on the model-based collaborative filtering technique using the single value decomposition and average least square prediction algorithms. The experimental results show that the Average strategy outperformed the Most Pleasure strategy for both prediction algorithms in terms of MAE, RMSE, and precision and recall metrics.

**Index Terms**—Model-based collaborative filtering, group modelling strategy, and group recommender system.

## I. INTRODUCTION

Recommender Systems (RS) are software tools that provide suggestions for items that are most likely of interest to a particular user [1], [2]. RSs help users during decision making by recommending the best alternative that suits their taste or needs. As a subclass of information screening system, RS seek to predict the users' ratings or preference on items, and then interrogatively recommend to users according to users' personal history activities and the features of the items [3]. Most RS focus on providing recommendation to a single user. However, nowadays, there are lots of situations and activities that users need to collaborate with others, and thus forming groups. It could be in the domain of vacation (i.e. a group of friends travelling together), music (a song that will be played randomly based on the group of user's preferences), and food (i.e., a group of colleagues having dinner together). In such situations, there is a need to recommend items to a group of users instead of a single user. Such systems are known as Group Recommender System (GRS). Hence, GRS has been researched in more recent years. GRS aims to make item recommendations that are "good" for a group of users as a whole, i.e. the items satisfy, as much as possible, the individual preferences of each group member [4].

Recommendation is a complex process, and many factors can influence the recommending results, among which, as

according to Bellogin *et al.* [5], similarity metric is undoubtedly an important one. Some traditional similarity algorithms are widely used in the existing RSs [6], [7] such as Pearson correlation similarity, Euclidean distance similarity, Cosine-based similarity, City Block similarity, Spearman correlation similarity and Tanimoto coefficient similarity. The core of Collaborative Filtering (CF) is to calculate similarities among users (user-based) or items (item-based). It recommends items that users with similar tastes and preferences liked in the past. Two users are considered similar if they have many 'liked' items in common. Thus, the system can use the preferences of similar users to predict the unrated items a user may like. Similarity functions are used to measure the similarity between users, which mainly rely on the user-item matrix values. However, in the case of the sparse data and the cold-start problem, good similarity measures sometimes are difficult to achieve, which thus deteriorate the recommendation accuracy [8].

This paper focuses on the performance of model-based CF algorithms for GRS. Model-based algorithms [9]-[11] use the collection of ratings to learn a predictive model, typically using some statistical or machine learning methods, which is then used to make predictions [6]. Our work falls into the k-Nearest Neighbours (kNN) technique, which is to find the neighbouring set of users who are most likely similar to each other, based on the user-ratings matrix. Similar users are clustered together to form a group. A group modelling aggregation strategy is then applied to the recently constructed group to assign 'representations' of the group. We explored two prediction algorithms for the GRS: The Alternating Least Square (ALS) and the Singular Value Decomposition (SVD).

Most of the study of GRS consider the group size on applying the group modelling strategy such as by Najjar and Wilson [12], Berkovsky and Freyne [13], Boratto & Carta [14] and Baltrunas *et al.* [15]. Baltrunas *et al.* [15] used simulated groups to compare aggregation strategies of ranked lists produced by a model-based collaborative filtering method using matrix factorization with gradient descent. While Najjar and Wilson [12] investigate the aspect of group homogeneity and size on the different aggregation methods used to predict a rating score for a group (mainly the MP, AV, and Least Misery) based on the memory-based CF algorithms. This led to our work in investigating the two aggregation strategies mentioned above in model-based CF. To the best of our knowledge, there is yet a paper that technically discussed and compared the performance of applying group modelling strategy (specifically GRS) based on the two aforementioned prediction algorithms (ALS and SVD). The aim of this paper is, therefore, twofold. First, is to compare the two most used group modelling approach in

Manuscript received September 25, 2019; revised January 11, 2020.

The authors are with the Center for Artificial Intelligent Technology, Universiti Kebangsaan Malaysia, Bangi, Selangor, Malaysia (e-mail: rosmamalmi@siswa.ukm.edu.my, shahrul@ukm.edu.my, lailatul.qadri@ukm.edu.my).

GRS, which are the Average strategy and the Most Pleasure strategy. Second, is to evaluate the prediction performance of the ALS and SVD prediction algorithms on both GRS group modelling strategies.

The structure of the paper is as follows. Section II would be elaborating the related studies in the area RS. Meanwhile, Section III describes the group formation and group modelling in the area of GRS. Section III presents the dataset and evaluation metrics used in the experimental results. Section V discusses the results, and finally, Section VI gives a description of the conclusions and future work.

## II. LITERATURE REVIEW

One of the aims of RSs is to reduce the information overload problem. Currently, RSs are considered as an essential part of various domains ranging from e-learning, movies, books, e-commerce and news to research articles. Methods in RSs can be generally categorized into three: the Collaborative Filtering method (CF), the Content-Based method (CB) and the Hybrid method. The CF method is the most widely used [9] and the most mature approach [16] for recommending items to users probably due to its ability in providing a recommendation without the needs of additional information such as the content of items. CF method also able to make recommendations on the basis of users' historical behaviours, such as clicking, browsing, and rating [17]. Algorithms in CF are based on the fact that similar users exhibit similar patterns of rating behaviours and similar items receive similar ratings. It assumed that if the ratings given by two users to the same items are similar, then the users might have a similar taste [18], [19]. Hence, the CF algorithm identifies a set of similar users called neighbours of a target user to whom an item will be recommended [20].

CF techniques can be classified into two: memory-based and model-based. The memory-based method performs recommendation by accessing the database directly, while model-based method uses the transaction data to create a model that can generate recommendation [21]. Model-based techniques are more preferred as memory-based approaches have scalability issues. One of the model-based CF approaches is based on matrix factorization, which has received greater exposure, mainly as an unsupervised learning method for latent variable decomposition and dimensionality reduction. Matrix factorization is widely used for RSs where it can deal better with scalability and sparsity. The aims of matrix factorization-based CF is first to reduce the dimension of the rating matrix; and then to discover possible features in the rating matrix which are used to drive recommendation [22]. ALS and SVD are among the algorithms classified under matrix factorization-based CF, which are being explored in this paper to provide recommendation for groups of users.

### A. Group Recommender System

The majority of studies focused on enhancing the accuracy of recommendation techniques for individual RS, with relatively few investigating to enhance the GRSs. A detailed survey of the state-of-the-art of GRS is provided by Masthoff [23]. Traditional or personal RSs apply three information resources (or input) [24] which are users, items and ratings

which are represented as follows:

$$\begin{aligned} \text{Users: } U &= \{u_1, u_2, \dots, u_n\}, \\ \text{Items: } I &= \{i_1, i_2, \dots, i_n\} \\ \text{Ratings: } R &\subseteq U \times I \rightarrow D. \end{aligned}$$

While GRSs can be defined as  $(G = \{m_1, \dots, m_r\} \subseteq U)$ , where  $m$  refers to users. GRSs based on Borrato *et al.* [25] is designed for contexts in which more than one person are involved in the recommendation process. GRSs usually generate group preferences through aggregating individual's ratings [26]. The recommendation made to the group in GRS can be described in equation (1), where  $G$  refers to the target group,  $I$  refer to the available set of items, and the  $\text{Prediction}(G, i_k)$  refers to the utility function for item  $i_k$  based on the group members of  $G$ .

$$\text{Recommendation}(G, I) = \underset{i_k \in I}{\text{arg max}} \text{Prediction}(G, i_k) \quad (1)$$

Approaches in GRSs generally follow a three-step process [22]:

- Group formation - identification of users with similar preferences as group members;
- Group modelling - aggregation of group members' preferences; and
- Prediction - prediction of unrated items.

GRSs algorithm suggests items to a group, by combining individual models that contain users' preferences [27]. The need for choosing a method of aggregation to generate recommendations is the key characteristic of group recommendation. Although different aggregation strategies differ in the way they manipulate and represent users' preferences, virtually all of them adopts one of the following three schemes: (1) aggregates a single set of individual recommendations, (2) builds a unique representation model for the group, or (3) aggregate the ratings/preferences for particular items.

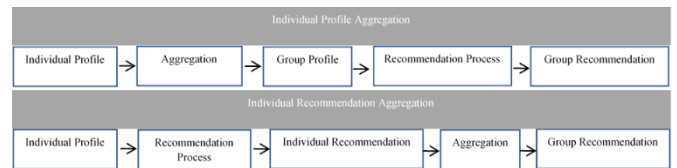


Fig. 1. Dominant approach in GRS.

Group recommendation has largely been studied in the context of CF [13], [15], [28]. Essential to note that as according to Jameson and Smyth [4] two dominant approaches for group recommendation are either aggregation of users' profiles or aggregation of recommendations. Aggregating profiles is where the different profiles of the group members are condensed into a single group profile; whereas aggregation of recommendations is where individual recommendations are merged to form a final list of recommendations. Fig. 1 illustrates the overall process of the two approaches in the GRSs. The following subsections provide a detail discussion of the three-step GRSs process.

Aggregating the profiles (preferences) may lead to group suggestions that lie outside the range of any individual recommendation list, which may be disorienting to the users and difficult to explain [29] nevertheless that lead to the issue of serendipity. Several experimental works show that neither

approach is better than the other in all scenarios [23], [30]. Also, these approaches rely on different aggregation strategies, which is each aggregation strategy has its own strengths and weaknesses.

Within approaches as explained above (also refer to Fig. 1), several aggregation strategies are used, such as least misery, most pleasure, average without misery or borda count, among others [23]. Berkovsky and Freyne [13] applied a memory-based recommendation approach in the recipe recommendation domain, comparing the performance of four recommendation strategies, including aggregated models and aggregated predictions. They reported better performance when aggregating the user profiles rather than aggregating individual user predictions. Work in [31] investigates how effective group recommendations for movies can be generated by combining the group members' preferences (as expressed by ratings) or by combining the group members' recommendations. Each approach is evaluated in a different method in RS; CF, CB, Hybrid method, SVD and most popular recommender (POP) as a baseline with various aggregation strategies in diverse of group size.

### B. Group Formation

In group recommendation scenarios, group formation process which involves group creation and group maintenance is a crucial and complex task that requires greater attention. According to Lu *et al.* [32], there are two types of group formation. The first is off-line, meant for the already existing groups such as a family to watch the television programmes, and the second is on-line, which are groups automatically formed by a system. Cantador and Castells [33], however, divide group formation into (i) intentionally, group clearly divided by group members; and (ii) non-intentionally, group automatically divided by a system. Boratto and Carta [25] classify group formation into four types, which are; established group, occasional group, random group and automatically detected group. They consider that once a user is clustered, strategies are tested in order to find the one that allows getting the best accuracy. In [14] they evaluate the group modelling strategies in a group recommendation scenario in which group are detected by clustering user.

Users' satisfaction issues when it comes to group recommendation have been discussed by Basu *et al.* [34]. Among the issues are forming groups such that the users in the formed groups are most satisfied with the suggested top-k recommendations. They apply some natural alternative (max-aggregation, min-aggregation, sum-aggregation, and weighted sum-aggregation) with varying number of users, number of items, and number of groups. However, it is beyond the scope of this paper to discuss in detail the group formation techniques. Table I provides a summary of the techniques used in existing approaches for group recommendation.

Based on Table I, clustering is the most used approach in group formation. The goal of clustering is to automatically find groups of instances (i.e., users) that are similar in a collection. The shape of the cluster depends on the type of similarity metrics or distance used. There are few available clustering techniques such as k-means, density-based, and hierarchical-based. Boratto and Carta [14], however,

highlighted that based on the study in [40], the k-means clustering algorithm is by far the most used clustering algorithm in recommender systems.

TABLE I: SUMMARY OF RELATED STUDIES FOR GROUP FORMATION IN GROUP RECOMMENDER SYSTEM

Research	Approach	Method	Similarity Metric
Mahyar <i>et al.</i> [35]	Centrality - User Based	a) Heuristic b) Optimization	a) Bayesian
Wu <i>et al.</i> (2016) [36]	Clustering (CF)	Markov Chain Monte Carlo	a) Popularity b) CF Item-Based (Jaccard Correlation) Kendall-Tau
Basu <i>et al.</i> [34]	Strategic	Greedy Algorithm	
Najjar & Wilson [12]	Neighborhood-Based CF	Group Threshold Matrix	Pearson
Boratto & Carta [14]	Clustering	K-Means	Pearson
Eslami <i>et al.</i> [37]	Clustering (online social network)	a) Markov b) OSLOM c) Louvain	a) Conductance b) Triad Participation Ratio (TPR)
Ntoutsis <i>et al.</i> [38]	Clustering	Hierarchical Agglomerative	
Saha & Getoor [39]	Proximity (online social network)	a) Topic-based Proximity b) Link-based proximity	Link Cardinality - regression tree - linear regression - SVM Multiclass

### C. Group Modelling

Group modelling strategies are inspired by Social Choice Theory and centered on modelling the achievement of consensus among the groups [41]. The aggregation process of group member's profile in the same group is usually called a group model/profile [27]. Data in the rating profile for all users in the group will be aggregated based on the aggregation strategies introduced by Masthoff [27]. Traditional recommendation for a single user tries to satisfy actual user needs; meanwhile, for a GRS based on the use of aggregation strategy, it tries to maximize the satisfaction of every user of the group. According to [42], aggregation strategies can be divided into three categories:

- Consensus-based strategies: considers the preferences of all group members. Among the strategies in this category are Average, Average without Misery, Fairness, and Multiplicative.
- Majority-based strategies: uses the most popular items among group members. The majority-based strategies allow satisfying the majority of the members of the group, even if the recommendation is extremely unsatisfying for the others. Among the strategies in this category is Plurality Voting, Borda Count and Copeland Rule.
- Borderline strategies: consider only a subset of items in individual profiles based on user roles or any other relevant criteria. In the Dictatorship strategy, for instance, a single-member imposes his taste for the rest of the group. Least Misery and Most Pleasure strategies consider only the lowest and highest level of interest, respectively, among the group members.

Masthoff [27] studies stated that people regularly used average strategy and least misery since they valued fairness

and preventing misery. While a study by [12] reported that, small-sized groups with high similarity among their members' average and most happiness perform the best. For larger size groups with high-similarity performs most happiness performs better. For the low and medium similarity groups, the average strategy has the best performance.

### III. MODEL-BASED CF AND GROUP MODELLING STRATEGIES

#### A. User Clustering

The more similar the user preferences are in the group, the better the group recommendations as mention by Pessemier *et al.* [31]. Thus, the accuracy of the group recommendations increases as the similarity between members of the group increases. In our work, we used k-Nearest Neighbour (kNN) as a method for clustering. Normally, kNN is a supervised learning algorithm for regression and classification. However, there is an obvious way to cluster via unsupervised nearest neighbours which is simply getting the nearest neighbours of a given point  $p$  either by taking all the neighbours in some ball around  $p$  with cut-off radius  $r$  or by taking the  $k$  nearest neighbours and returning them as the cluster.

The traditional CF approach is commonly referred to as neighbourhood-based and relies on the fact that each person belongs in a larger group of similarly behaving individuals. The main step of the kNN algorithm is calculating users' similarity to find the most similar  $k$  number of nearest neighbours to a given user (refer to an as active user). We use the cosine similarity to find the similarities between users. The similarity of two user vectors is used to specify the nearest neighbours of an active user. A cosine similarity metric is ideal for sparse vectors. The cosine similarity between two vectors of  $A$  and  $B$  is defined as in equation 2.

$$\text{similarity}(A, B) = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2 \sum_{i=1}^n B_i^2}} \quad (2)$$

#### B. Algorithm in Model-Based CF

Model-based use RS information to create a model that generates recommendations. Model-based approaches (such as machine learning and data mining algorithms) are used to design and develop models that allow the system to learn complex patterns based on training data and then provide predictions based on the models, thus learned [43]. Among the most widely used models, as stated in [44], are Bayesian classifiers, neural networks, fuzzy systems, genetic algorithms, latent features and matrix factorization, among others. We focus on the matrix factorization area that applies SVD and ALS algorithm.

SVD is closely related to the Matrix Factorization method. It decomposed the original rating matrix  $R$  by  $A = U \times S \times V^T$ , where  $U$  and  $V$  are the left and right singular vectors and  $S$  is the diagonal matrix. The first  $k$  singular values are chosen to reduce the dimension of the diagonal matrix  $S$ . The reconstruction of  $R_k = U_k \times S_k \times V_k^T$  gives the *rank-k* matrix which is the closest approximation of original rating matrix [45].

While the ALS algorithm creates a matrix of all users

versus all movies. It fills in the probable (predicted) ratings, based on similarities between user ratings. The algorithm uses the least square computation to minimize the estimation errors and alternates between solving for movie factors and solving for user factors.

#### C. Aggregation Strategies

In this paper, two strategies have been selected, which are the Average (AV) and the Most Pleasure (MP) strategies.

##### 1) Average

It is known as a basic group aggregation strategy that adopts equal influence among group members and calculates the average rating of the group members for any given item as the predicted rating. Because this method corresponds to one of how a group of people naturally make choices [41], we used this aggregation method for the experiments as discussed in Section 4.0 and 5.0. Let  $n$  be the number of users in a group and  $r_{ai}$  be the rating of user  $a$  for item  $i$ , then the group rating for item  $i$  is computed as follows (Equation 3);

$$Gr_i = \frac{\sum_{a=1}^n r_{ai}}{n} \quad (3)$$

Table II describes the example scenario of the application of AV Strategy for four users and five items.

	Item A	Item B	Item C	Item D	Item E
User 1	4	3	5	4	1
User 2		2		3	
User 3	3		3	4	2
User 4	2		4	4	2
Group Rating	3	2.5	4	3.75	1.67

##### 2) Most pleasure

The MP strategy is also known as Most Happiness and Most Respected Person. This aggregation strategy assumes that a 'happy' group is based on their happiest member. It assumes that the other members are satisfied with the highest rated items among the members, and the rated items represent the group preference. This strategy tries to recommend alternately the items that one group member really likes, thereby not considering the preferences of other members [31]. Equation 4 represents the equation for the MP strategy, where  $r_{ai}$  refers to ratings given by user  $a$  on item  $i$ .

$$Gr_i = \max_a r_{ai} \quad (4)$$

Table III represents an example of the MP strategy.

	Item A	Item B	Item C	Item D	Item E
User 1	4	3	5	4	1
User 2		2		3	
User 3	3		3		2
User 4	2		4	4	
Group Rating	4	3	5	4	2

## IV. EVALUATION

As mentioned in section 1, the aim of this study is twofold.

First, is to compare the two most used group modelling approach in GRS, which are the Average strategy and the Most Pleasure strategy. Second, is to evaluate the prediction performance of the two prediction algorithms (ALS and SVD) on both GRS group modelling strategies. For group formation, we used the kNN clustering algorithm, whereby the similarity between users is measured by the cosine similarity metric.

A. Dataset

The Movielens 1M dataset was employed to evaluate the effectiveness of the explored algorithms. The dataset includes 6040 users and 3952 movies with 1,000,209 ratings. In the data sets, each person has rated at least 20 movies. The user profile includes age, gender, and profession. The movie includes 19 types of genres. Movielens 1M dataset is widely used by many researchers and seems to establish itself as a standard dataset in CF evaluation.

B. Evaluation Metrics

Performance score can be improved by implementing k-Fold Cross-Validation (where k refers to the number of groups that a given data sample is to be split). Hence, we also used this method to enhance the performance score for the evaluation. To estimate the performance of GRS, the Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), precision and recall are among the most popular ones. According to [29], the metrics evaluating RS can be roughly classified into two categories: prediction accuracy and classification accuracy. RMSE (Equation 5) and MAE (Equation 6) are mainly used to evaluate the prediction accuracy [46], while Precision (Equation 7) and Recall (Equation 8), as well as F1-Score (Equation 9), are used to evaluate the quality of top N recommendation. The equation of RMSE is as follows;

$$RMSE = \sqrt{\frac{\sum_{i=0}^n (r_{ui} - p_{gi})^2}{n}} \tag{5}$$

$$MAE = \frac{1}{n} \sum_{i=0}^n |d_i| - |\hat{d}_i| \tag{6}$$

RMSE is useful in finding out the ability of a CF algorithm to generalize and highlights larger errors. While MAE is to measure the effect on prediction stability.

However, RMSE and MAE do not really reflect the real user experience. Thus, according to McLaughlin & Herlocker [47] precision and recall reflect the real user experience better than RMSE and MAE do because in most cases, users actually received ranked lists from a recommender instead of predictions for ratings of specific items. Both precision and recall are computed as fractions of  $hits_u$  which is the number of correctly recommended relevant items for user  $u$ . The equations for precision (P) and recall (R) are as follows respectively, where  $recset_u$  are the recommended items for user  $u$  and  $testset_u$  refers to hits owing to the testing set size.

$$P_u = \frac{|hits_u|}{|recset_u|} \tag{7}$$

$$R_u = \frac{|hits_u|}{|testset_u|} \tag{8}$$

$$F1\text{-score} = \frac{2 \times P_u \times R_u}{P_u + R_u} \tag{9}$$

V. RESULT AND DISCUSSION

To find the optimum  $k$  value for clustering the users to a group, we apply the elbow method. Based on the result of  $k = 5, 10, 15, 20, 30,$  and  $50,$  the optimum value for  $k$  is 10 as it stops the result from overfitting and underfits as shown in Fig. 2. For that reason, we choose to cluster ten users for a group. Note that the same user can be in several different groups, and we plot only data for a sample of 25 groups.

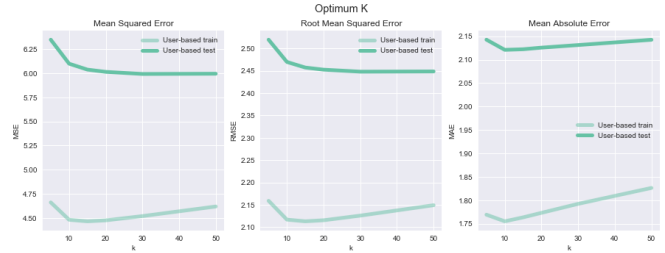


Fig. 2. Optimum k value for clustering.

User similarity based on the cosine similarity, which represents five initial users and five initial items is shown in Table IV. It can be referred to as the higher the similarity, the closer the neighbour.

TABLE IV: USER SIMILARITY WITH COSINE SIMILARITY METRIC

	U1	U2	U3	U4	U5
U <sub>1</sub>	1.000000	0.096382	0.120610	0.132455	0.090158
U <sub>2</sub>	0.096382	1.000000	0.151479	0.171176	0.114394
U <sub>3</sub>	0.120610	0.151479	1.000000	0.151227	0.062907
U <sub>4</sub>	0.132455	0.171176	0.151227	1.000000	0.045094
U <sub>5</sub>	0.090158	0.114394	0.062907	0.045094	1.000000

By implementing kNN method with  $k=10,$  ten users will be clustered into a group. An example of a cluster is as shown in Table V. Then, group modelling based on aggregation strategy will be applied to generate ratings that represent the group. As mentioned earlier, the AV and MP are the two strategies being tested in our experiments.

TABLE V: EXAMPLE OF CLUSTER

User ID	Item ID									
	i1	i2	i3	i4	i5	i6	i7	i8	i9	i10
5014	5					5				3
168	5		5		4	1				2
2785						4				4
325	3									
1779	3	3	2			2				
4057	3	2				4				2
3716	5									
2637	5	3				3				1
5366	5	2	3	2		4		2		3
3723	5		3		3	4				

Table VI and VII respectively illustrate the results of group ratings for the MP and AV strategies. The MP strategies give a big round number for group rating as it considers the



highest rating. Compared to the rating value as group rating for AV strategy, the number are mostly in decimal value with lower rating value than MP strategies as it takes the average number of group members' preferences. Fig. 3 depicts the differences of rating distribution after applying AV and MP strategy by violin plot chart. These differences would be highly impacted to the item recommendation as will be discussed further below.

TABLE VI: GROUP RATING FOR MP STRATEGY

User ID	Item ID									
	i1	i2	i3	i4	i5	i6	i7	i8	i9	i10
5014	5					5				3
168	5		5		4	1				2
2785						4				4
325	3									
1779	3	3	2			2				
4057	3	2				4				2
3716	5									
2637	5	3				3				1
5366	5	2	3	2		4		2		3
3723	5		3		3	4				
Group Rating	5	3	5	2	4	5		2		4

TABLE VII: GROUP RATING FOR AV STRATEGY

User ID	Item ID									
	i1	i2	i3	i4	i5	i6	i7	i8	i9	i10
5014	5					5				3
168	5		5		4	1				2
2785						4				4
325	3									
1779	3	3	2			2				
4057	3	2				4				2
3716	5									
2637	5	3				3				1
5366	5	2	3	2		4		2		3
3723	5		3		3	4				
Group Rating	4.3	2.5	3.3	2	3.5	3.4		2		2.5

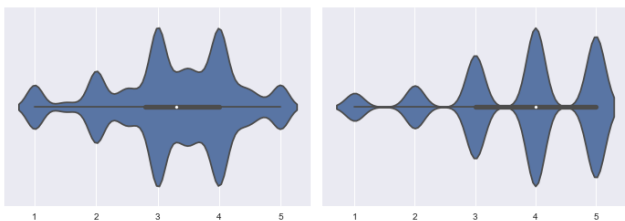


Fig. 3. Distribution of rating for AV strategy (left) and MP strategy (right).

In this paper, we focus on the top N recommendation scenario, in which a recommendation model suggests a list of top N most appealing items to the group. Group recommendations are offered to group members. The result for the top-5 movie recommendation for the Cluster 1 (C1) based on the aggregation of MP strategy of ALS and SVD algorithm can be referred in Table VIII. We notice that none of the recommended movies is similar between both recommendation algorithms. However, both results show similar patterns in terms of movie genre such as 'Action', 'Drama', and 'Thriller'. For example, the movies 'North by

Northwest (1959)' and 'Mission: Impossible 2 (2000)' recommended using the ALS algorithm; and the movies 'The Godfather (1972)' and 'Chinatown (1974)' recommended using the SVD algorithm are all belong to the genre.

Even we do not consider the genre feature for movie prediction in our work, ALS and SVD as matrix factorization-based, also seeing the item attributes (this could be movie characteristic such as movie genre, movie director, etc.) other than user preferences to find a good movie recommendation for the user. The principle of the matrix factorization-based itself, which is applied dot-matrix multiplication between cluster preferences and movie characteristic as well to get the additional and rich information. Thus, we can see that from the result of movie recommendation, the system enables to provide recommendation for C1 within a similar genre, even not exactly the same movies.

According to Pessemier *et al.* [31], the approach of aggregating users' preferences increases the chance of recommending serendipitous items as compared to aggregating individual recommendations approaches. Such a hypothesis seems to apply to our experiments as exhibited in the MP strategy for ALS algorithm, whereby the movie 'West Side Story (1961)' which is categorized in musical and romance genre 'surprisingly' being recommended if we consider the similar genre of top N recommendation produced.

TABLE VIII: TOP-5 MOVIE RECOMMENDATION OF GRS BASED ON THE MP AGGREGATION STRATEGY

Algorithm	Movie Title
ALS	Sense and Sensibility (1995)   Drama   Romance
	North by Northwest (1959)   Drama   Thriller
	Big Sleep, The (1946)   Film-Noir   Mystery
	West Side Story (1961)   Musical   Romance
	Mission: Impossible 2 (2000)   Action   Thriller
SVD	Godfather, The (1972)   Action   Crime   Drama
	Cider House Rules, The (1999)   Drama
	Parenthood (1989)   Comedy   Drama
	Chinatown (1974)   Film-Noir   Mystery   Thriller
	Monty Python's Life of Brian (1979)   Comedy

TABLE IX: TOP-5 MOVIE RECOMMENDATION OF GRS BASED ON THE AV AGGREGATION STRATEGY

Algorithm	Movie Title
ALS	Three Colors: Red (1994)   Drama
	Akira (1988)   Adventure   Animation   Sci-Fi   Thriller
	Smilla's Sense of Snow (1997)   Action   Drama   Thriller
	Dark City (1998)   Film-Noir   Sci-Fi   Thriller
	Boondock Saints, The (1999)   Action   Comedy
SVD	Sunset Blvd. (a.k.a. Sunset Boulevard) (1950)   Film-Noir
	Sling Blade (1996)   Drama   Thriller
	Streetcar Named Desire, A (1951)   Drama
	Family Thing, A (1996)   Comedy   Drama
	Harold and Maude (1971)   Comedy

On the other hand, Table IX shows the recommended movies for cluster C1 based on the AV strategy for the ALS and SVD algorithms. The AV strategy represents the group rating by averaging the ratings given by all members. Thus, it treats all group members as equally important. However, there exists a situation that there are some members who may have more influence on the group recommendations than the other members. For example, in the case where there is only one member in a cluster that provides ratings to items, as illustrated in Table VI and VII for item i8. Both MP and AV strategies produce the same group rating for item i8. So, this

situation might be one of the reasons that MP and AV strategy with SVD and ALS prediction algorithm produce non-identical movie recommendation but similar movie genres. We can see from the output presented in Table VIII and Table IX, both MP and AV strategies recommend different movies, but the majority of movies being recommended belong to similar genres.

We implement k-Fold Cross-Validation (applying 5-Fold Cross-Validation in this experiment) technique to get a better performance score. We randomly pick 70% of the rating data as our training set.

TABLE X: PERFORMANCE SCORE FOR GROUP MODELLING STRATEGIES

Algorithm	Evaluation Performance	Group Modelling Strategies	
		MP Strategy	AV Strategy
ALS	RMSE	1.0319	0.8761
	MAE	0.8090	0.6633
	Precision@5	0.8320	0.8960
	Recall@5	0.0486	0.0603
	F1@5	0.0918	0.1130
SVD	RMSE	0.9673	0.8244
	MAE	0.7609	0.6534
	Precision@5	0.9200	0.9440
	Recall@5	0.0460	0.0616
	F1@5	0.0878	0.1156

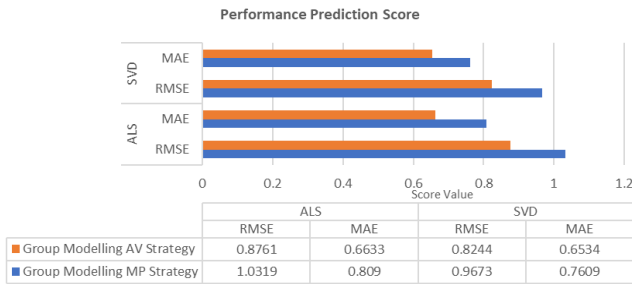


Fig. 4. Performance prediction score of AV Strategy and MP strategy based on ALS and SVD algorithm.

The strategy used to model groups significantly influence the accuracy of the group recommendations [48]. The prediction accuracy score of the two recommendation algorithms is compared between the AV and MP group strategies. Lower score RMSE and MAE indicates that the predicted ratings are closer to actual values. The results in Table X and Fig. 6, show that for ALS algorithm, the values of RMSE and MAE for the MP strategy are 1.0319 and 0.8090, respectively. Meanwhile, the AV strategy produces a better performance, which is 0.8761 for RMSE and 0.6633 for MAE. Based on the result in Table X and Fig. 6 for the SVD algorithm as well, we can perceive that AV strategy produces better prediction as compared to the MP strategy. The AV strategy with SVD algorithm resulted in the lowest RMSE and MAE, thereby outperforming the MP strategy for both ALS and SVD.

The MP strategy, which is categorized under the borderline strategies, as discussed in Section II.C, considers the highest rating among members and ignore other preferences. While the AV strategy which is the consensus-based strategies takes into account all of the members' preferences. Averaging individual preferences return group preferences that equally consider each user. If a strategy that only consider part of the group preference, the accuracy of the group recommendations will be affected [48]. This differences of the preferences of the item as a final group rating could lead

to the different prediction of accuracy value thus come out with various movie recommendation. The aggregation strategies compared in this work, the result proves the statement by Boratto and Carta [48], that it can be noticed that the strategies that advantage a user (i.e., Least Misery and Most Pleasure) do not produce accurate models.

However, the aggregation methods used may also influenced by the used recommendation algorithms. For example, the work by Pessemier *et al.* [31] shows that the CB recommendation approach produces better results with the MP strategy. On the hand, the Hybrid recommendation approach produces more accurate results with AV strategy.

The aggregation method that produces the most accurate group recommendations depends on the used algorithm and grouping strategy [31]. In addition, a study by Pessemier *et al.* [31], reported that CB combination with profile aggregation approach produces the most accurate group recommendations if the MP strategy is used. While the Hybrid method with recommendation aggregation approach generates the most accurate recommendations if the AV strategy is used. Interesting to note that our work who fall into the approach of profile aggregation with CF, produce better accuracy with the AV strategy as compared to the MP strategy.

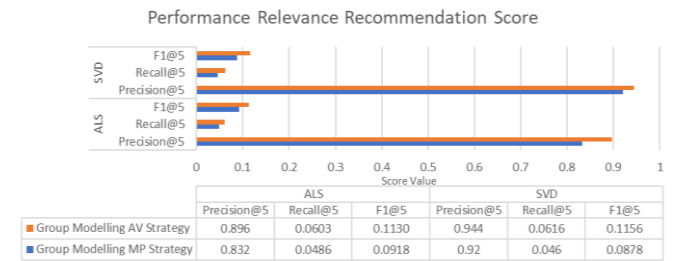


Fig. 5. Performance relevance recommendation score of AV strategy and MP strategy based on ALS and SVD algorithm.

Table X and Fig. 5 also shows the recommendation performance between the MP and AV strategy for both recommendation algorithms in terms of precision and recall. The AV strategy produces better results for both algorithms, with 89.6% precision and 6.03% recall for the ALS algorithm as compared to 83.2% and 4.86% of the precision and recall values respectively for the MP strategy. Similar patterns of performance exhibited in the SVD algorithm whereby the AV strategy able to achieve 94.4% and 6.16% for precision and recall respectively as compared to 89.7% and 6.03% for the MP strategy. Based on the F1 score, overall the AV strategy produces better recommendation as compared to the MP strategy.

## VI. CONCLUSION AND FUTURE WORKS

This paper has presented the GRS by applying individual profile aggregation on group modelling strategy, which are the MP strategy and the AV strategy. In this work, we compared the performance score based on the use of ALS and SVD prediction algorithm. We emphasized the use of accuracy metric and relevance recommendation to compute performance evaluation score. Comparing both algorithms of ALS and SVD, we found that AV strategy gives better result in terms of prediction accuracy and relevant recommendation

at top-5. Until now, group recommendations are still less explored as compared to individual recommendations [27]. Compared to other studies that take into consideration of other criteria on the aggregation strategy such as group size, group homogeneity, group consensus concerning relationship [24], and group member personality with social relationship [28], our work solely focus on the simple grouping strategy for movie recommendation as the main is mainly on evaluating the performance of the MP and AV strategies. However, the abovementioned criteria are worthy to be look into in the near future. Our near future works, includes exploring other clustering techniques such as k-Means or k-Medoids to produce better user clustering. Considering other group modelling strategies such as the approval voting and the average without misery are also part of our near future works.

#### CONFLICT OF INTEREST

“The authors declare no conflict of interest”

#### AUTHOR CONTRIBUTIONS

R. Mat Nawi; S. A. Mohd Noah, and L. Q. Zakaria conceived of the presented idea. R. Mat Nawi; S. A. Mohd Noah, designed and performed the experiments, analysed data and co-wrote the paper. S. A. Mohd Noah, and L. Q. Zakaria involved in supervised the work and assist in the technical details.

#### ACKNOWLEDGMENT

This research is partially supported by the Universiti Kebangsaan Malaysia under the Grand Challenge Fund DCP-2017-002/3.

#### REFERENCES

- [1] A. S. Ghabayen and S. A. M. Noah, “Using tags for measuring the semantic similarity of users to enhance collaborative filtering recommender systems,” *International Journal on Advanced Science, Engineering and Information Technology*, vol. 7, no. 6, pp. 2063-2070, 2017.
- [2] N. A. Osman, S. A. M. Noah, and M. Darwich, “Contextual sentiment-based recommender system to provide recommendation in the electronic products domain,” *International Journal of Machine Learning and Computing*, vol. 9, no. 4, pp. 425-431, 2019.
- [3] N. I. Y. Saat, S. A. M. Noah, and M. Mohd, “Towards serendipity in for content-based recommender systems,” *International Journal on Advanced Science Engineering and Information Technology*, vol. 8, pp. 1763-1769, 2018.
- [4] A. Jameson and B. Smyth, “Recommendation to groups,” *LNCS*, vol. 4321, pp. 596–627.
- [5] A. Bellog  and A. P. de Vries, “Understanding similarity metrics in neighbor-based recommender systems,” in *Proc. the 2013 Conference on the Theory of Information Retrieval*, 2013, vol. 13.
- [6] 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*, pp. 743-749, 2005.
- [7] A. S. Ghabayen and S. A. Noah, “Exploiting social tags to overcome cold start recommendation problem,” *Journal of Computer Science*, vol. 10, no. 7, pp. 1166-1173, 2014.
- [8] S. Fan, H. Yu, and H. Huang, “An improved collaborative filtering recommendation algorithm based on reliability,” in *Proc. 3rd IEEE International Conference on Cloud Computing and Big Data Analysis*, 2018.
- [9] J. S. Breese, D. Heckerman and C. Kadie, “Empirical analysis of predictive algorithms for collaborative filtering,” in *Proc. the Fourteenth Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann Publishers Inc., 1998, pp. 43–52.
- [10] K. Goldberg, T. Roeder, D. Gupta and C. Perkins, “Eigentaste: A constant time collaborative filtering algorithm,” *Information Retrieval*, vol. 4, no. 2, pp. 133–151, 2001.
- [11] D. Pavlov and D. M. Pennock, “A maximum entropy approach to collaborative filtering in dynamic, sparse, high dimensional domains,” *Neural Information Processing Systems*, pp. 1441–1448, 2002.
- [12] N. A. Najjar and D. C. Wilson, “Evaluating group recommendation strategies in memory-based collaborative filtering,” in *Proc. the 5th ACM Recommender System International Conference Workshop on Human Decision Making in Recommender System*, Chicago, USA, 2011.
- [13] S. Berkovsky and J. Freyne, “Group-based recipe recommendations: analysis of data aggregation strategies,” in *Proc. the 4th ACM Recommender Systems Conference*, Barcelona, Spain, September 2010, pp. 111–118.
- [14] L. Boratto and S. Carta, “Using collaborative filtering to overcome the curse of dimensionality when clustering users in a group recommender system,” in *Proc. the 16th International Conference on Enterprise Information Systems*, 2014, vol. 2, pp. 564-572.
- [15] L. Baltrunas, T. Makcinskas and F. Ricci, “Group recommendations with rank aggregation and collaborative filtering,” in *Proc. the 4th ACM Recommender Systems Conference*, 2010, pp. 119–126.
- [16] Y. Koren, “Factorization meets the neighborhood: a multifaceted collaborative filtering model,” in *Proc. the 14th ACM SIGKDD international Conference on Knowledge Discovery and Data Mining*, New York, NY, USA, 2008, pp. 426–434.
- [17] J. Zhao and G. Sun, “Detect user’s rating characteristics by separate scores for matrix factorization technique,” *Symmetry*, vol. 10, 2018.
- [18] N. A. Osman and S. A. M. Noah, “Sentiment-based model for recommender systems,” in *Proc. 2018 Fourth International Conference on Information Retrieval and Knowledge Management*, pp. 85-90.
- [19] L. M. de Campos, J. M. Fern andez-Luna, J. F. Huete and M. A. Rueda-Morales, “Combining content-based and collaborative recommendations: A hybrid approach based on bayesian networks,” *Int. J. Approx. Reasoning*, vol. 51, pp. 785–799, 2010.
- [20] A. M. Jorge, J. Vinagre, M. Domingues, J. Gama, C. Soares, P. Matuszyk and M. Spiliopoulou, *Scalable Online Top-N Recommender Systems*, pp. 3–20, 2017.
- [21] J. Bobadilla, F. Ortega, A. Hernando and A. Guti erez, *Recommender systems survey*, *Knowledge-Based Systems*, vol. 46, pp. 109-132, 2013.
- [22] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. T. Riedl, “Application of dimensionality reduction in recommender system - A case study,” in *Proc. ACM WEBKDD Workshop*, 2000.
- [23] J. Masthoff, “Group recommender systems: Aggregation, satisfaction and group attributes,” *Recommender Systems Handbook*, Springer, pp. 743–776, 2015.
- [24] J. Castro, F. J. Quesada, I. Palomares, and L. Martnez, “A consensus-driven group recommender system,” *Journal International Journal of Intelligent Systems*, vol. 30, issue 8, pp. 887-906, August 2015.
- [25] L. Boratto and S. Carta, “State-of-the-art in group recommendation and new approaches for automatic identification of groups,” *Information Retrieval and Mining in Distributed Environments*, vol. 324, pp. 1–20, 2011.
- [26] M. Kompan and M. Bielikova, “Group recommendations: Survey and perspectives,” *Computing and Informatics*, vol. 33, no. 2, pp. 446–476, 2014.
- [27] J. Masthoff, “Group recommender systems: Combining individual models,” *Recommender Systems Handbook*, pp. 677–702, 2011.
- [28] L. Q. Sanchez, J. A. R. Garcia, and B. D. Agudo, “Group recommendation methods for social network environments,” in *Proc. 3rd Workshop on Recommender Systems and the Social Web*, 2011.
- [29] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, “Evaluating collaborative filtering recommender systems,” *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 5–53, 2004.
- [30] F. Ortega, A. Hernando, J. Bobadilla, and J. H. Kang, “Recommending items to group of users using matrix factorization based collaborative filtering,” *Inf. Sci.*, vol. 345, pp. 313–324, Jun. 2016.
- [31] T. D. Pessemier, S. Dooms, and L. Martens, “Comparison of group recommendation algorithms,” *Multimedia Tools Appl.*, vol. 72, no. 3, pp. 2497–2541, Oct. 2014.
- [32] J. Lu, D. Wu, M. Mao, W. Wang, and G. Zhang, “Recommender system application developments: A survey,” *Journal Decision Support System*, vol. 74, issue C, pp. 12-32, June 2015.
- [33] I. Cantador and P. Castells, “Group recommender systems: New perspectives in the social web,” *Recommender Systems for the Social Web*, vol. 32, 2012.



- [34] S. Basu Roy, L. V. S. Lakshmanan, and R. Liu, "From group recommendation to group formation," in *Proc. the 2015 ACM SIGMOD International Conference on Management of Data*, 2015, pp. 1603-1616.
- [35] H. Mahyar, E. K. Ghalebi, S. M. Morshedi, S. Khalili, R. Grosu, and A. Movaghar, "Centrality-based group formation in group recommender system," in *Proc. the 26th International Conference on World Wide Web Companion*, Perth, Australia, 2017, pp. 1187-1196.
- [36] Y. Wu, X. Liu, M. Xie, M. Ester, and Q. Yang, "CCCF: Improving collaborative filtering via scalable user-item co-clustering," in *Proc. the Ninth ACM International Conference on Web Search and Data Mining*, 2016, pp. 73-82.
- [37] M. Eslami, A. Aleyasen, R. Z. Moghaddam, and K. Karahalios, "Friend grouping algorithms for online social networks: Preference, bias, and implications," in *Proc. International Conference on Social Informatics*, 2014.
- [38] E. Ntoutsi, K. Stefanidis, K. Norvag, and H. Kriege, "Fast group recommendations by applying user clustering," *LNCS 7532*, pp. 126-140.
- [39] B. Saha and L. Getoor, "Group proximity measure for recommending groups in online social network," in *Proc. the 2nd SNA-KDD Workshop '08 (SNA-KDD '08)*, August 24, 2008.
- [40] X. Amatriain, A. Jaimes, N. Oliver, and J. M. Pujol, "Data mining methods for recommender systems," *Recommender Systems Handbook*, pp. 39-71, 2011.
- [41] J. Masthoff, "Group modeling: Selecting a sequence of television items to suit a group of viewers," *User Modeling and User-Adapted Interaction*, vol. 14, no. 1, 2004.
- [42] C. Senot, D. Kostadinov, M. Bouzid, J. Picault and A. Aghasaryan, "Evaluation of group profiling strategies," in *Proc. the Twenty-Second International Joint Conference on Artificial Intelligence*, Barcelona, 2011, pp. 2728-2733.
- [43] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Adv. Artificial Intelligence*, 2009.
- [44] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowledge-Based Systems*, vol. 46, pp. 109-132, 2013.
- [45] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Incremental singular value decomposition algorithms for highly scalable recommender systems," in *Proc. Fifth International Conference on Computer and Information Science*, Citeseer, 2002, pp. 27-28.
- [46] C. Zhang, M. Gartrell, T. P. Minka, Y. Zaykov, and J. Guiver, "GroupBox: A generative model for group recommendation," *Tech. Rep. MSR-TR-2015-61*, Microsoft Research, 2015.
- [47] M. R. McLaughlin and J. L. Herlocker, "A collaborative filtering algorithm and evaluation metric that accurately model the user experience," in *Proc. the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2004.
- [48] L. Boratto and S. Carta, "Modeling the preferences of a group of users detected by clustering: a group recommendation case-study," in *Proc. the 4th International Conference on Web Intelligence, Mining and Semantics*, 2014.

Copyright © 2020 by the 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 ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).



**Rosmamalmi Mat Nawi** is a PhD student within the Center for Artificial Intelligence Technology at Universiti Kebangsaan Malaysia. She received her bachelor's degree in computer science from Universiti Teknikal Malaysia Melaka, and master's degree in internet computing from Universiti Pendidikan Sultan Idris. Her thesis is concerned with the applying semantic web technologies in group recommender

system.



**Shahrul Azman Mohd Noah** received the BSc with honors in mathematics from the Universiti Kebangsaan Malaysia in 1992, MSc and PhD degrees in information studies from the University of Sheffield, UK, in 1994 and 1998, respectively. He is a professor in the Center for Artificial Intelligence Technology, Universiti Kebangsaan Malaysia and currently heads the knowledge technology research group. His research interests include information retrieval and ontology with special emphasis on semantic search and recommender systems. He has published more than 200 research articles in these areas. Prof. Mohd Noah is currently the chair of the Persatuan Capaian Maklumat dan Pengurusan Pengetahuan (PECAMP), members of the International Association for Ontology and its Applications (IAOA) and IEEE Computer Science Society associations.



**Lailatul Qadri Zakaria** is a lecturer of Natural Language Processing at Faculty of Information Science and Technology, University Kebangsaan Malaysia. She has a bachelor and master degree in information science from Universiti Kebangsaan Malaysia and she obtained her doctor's degree from University of Southampton, United Kingdom. Her research interest is in natural language processing, ontology development and semantic web technologies.