DOI: https://doi.org/10.15276/hait.07.2024.3
UDKC 004.42

Reaching consensus in group recommendation systems

Anastasiia A. Gorbatenko[1)]
ORCID: https://orcid.org/0000-0002-5165-5168; nastya000511@gmail.com
Mykola A. Hodovychenko[1)]
ORCID: https://orcid.org/0000-0001-5422-3048; hodovychenko@od.edu.ua. Scopus Author ID: 57188700773
[1)] Odessa Polytechnic National University, 1, Shevchenko Ave. Odessa, 65044, Ukraine

ABSTRACT

Conventional group recommender systems fail to take into account the impact of group dynamics on group recommendations, such as the process of reconciling individual preferences during collective decision-making. This scenario has been previously examined in the context of group decision making, specifically in relation to consensus reaching procedures. In such processes, experts engage in negotiations to determine their preferences and ultimately pick a mutually agreed upon option. The objective of the consensus procedure is to prevent dissatisfaction among group members about the suggestion. Prior studies have tried to accomplish this characteristic in group recommendation by using the minimal operator for the process of aggregating recommendations. Nevertheless, the use of this operator ensures just a minimal degree of consensus on the proposal, but it does not provide a satisfactory level of agreement among group members over the group recommendation. This paper focuses on analyzing consensus reaching procedures in the context of group recommendation for group decision making. The goal of the study is to use consensus reaching processes to provide group recommendations that satisfy all members of the group. Additionally, study aims to enhance group recommender systems by ensuring an acceptable level of agreement among users regarding the group recommendation. Therefore, group recommender systems are expanded by including consensus reaching mechanisms to facilitate group decision making. In the context of group decision making, a collective resolution is reached by a group of persons, who may be specialists, from a pool of options or potential solutions to the issue at hand. To do this, each specialist obtains their preferences about each possibility. The conventional selection techniques for group decision-making difficulties fail to include the possibility of dissent among experts over the chosen choice. This issue is alleviated by using consensus-building techniques, in which a substantial degree of agreement is attained prior to picking the ultimate decision. To facilitate alignment of experts' tastes, they repeatedly modify them to increase their proximity. Prior to making collective choices, it is sometimes necessary to establish a certain degree of consensus. Thus, this paper presents a group recommendation architecture that utilizes automated consensus reaching models to provide accepted suggestions. More specifically, we are considering the minimal cost consensus model and the automated consensus support system model that relies on input. The minimal cost consensus model calculates the collective suggestion of a group by adjusting individual preferences based on a cost function. This is achieved via the use of linear programming. The feedback-based automated consensus support system model mimics the interaction between group members and a moderator. The moderator offers adjustments to individual suggestions in order to bring them closer together and achieve a high degree of agreement before generating the group recommendation. Both models are assessed and contrasted with baseline procedures in the testing.

Keywords: Recommender system; machine learning; neural networks; deep learning; classification; information filtering system; information system

***For citation*: Gorbatenko A. A., Hodovychenko M. A. “Reaching consensus in group recommendation systems”.** *Herald of Advanced Information Technology*. **2024; Vol. 7 No. 1: 36–47. DOI: https://doi.org/10.15276/hait.07.2024.3**

INTRODUCTION, FORMULATION OF THE PROBLEM

As more people become digitally connected in today's society, there is an abundance of information available online. The proliferation of information has made it more difficult for users to locate the information they need. The purpose of internet search engines such as Google and Bing are to provide relevant and helpful information to users, but they are also becoming less important due to the challenge of locating relevant information among thousands of results [1].

Personalized web apps are needed to solve this issue since they gather the most important and useful data from many sources. Web customization offers a plethora of applications, the most noteworthy of which is the recommender system, which facilitates easy content recognition and information access in a manner that is worthwhile and acceptable for consumers [2].

A typical recommender system’s general goal is to observe user behavior, forecast an item's rating based on that information, and then recommend goods with the highest possible rating that consumers could find appealing [3].

One of the most common human tasks in everyday life is decision-making. Group decision-making procedures, in which a number of users collaboratively choose a single, shared answer from among many possible options, are becoming more and more necessary [4].

In requirements engineering settings, for example, members of a software development team must collaboratively choose one or more needs from a list of possible requirements to be included in the next release. In these scenarios, recommender systems that take into account each member's taste and preferences for the user group are referred to as group recommender systems [5].

In the realm of recommender systems, research in this area is ongoing. A plethora of group recommender systems have been created recently to address the difficulties involved in formulating suggestions for a group of members. The general goal of a traditional recommender system is to analyze user behavior, forecast item ratings based on that information, and then suggest goods with the highest rating that consumers may find appealing [6].

Group recommender system's primary responsibility is to identify each user's preferences and then identify a compromise that the group as a whole can agree upon. The goal of group suggestion is to create and compile each user's unique preferences.

The three main methods for creating the preference aggregation are as follows:

(a) combining the individual suggestions;

(b) aggregating the individual ranks.

(c) building a model of group preferences.

Certain group recommender systems create the group profile by giving each member of the group the same weight and avoiding any interactions between them. GRSK, Let's Browse, Polylens, Intrigue, MusicFX, and The Collaborative Advisory Travel System are a few examples of this kind of group recommender systems [7].

A collection of recommendations in conventional group recommender systems may be directed by aggregation techniques like majority, average, greatest joy, and least pain. These tactics, however, don't always result in group suggestions that are highly agreed upon by all participants. It is possible that some members of the group will not agree with the solution that was selected. Integrating a consensus-achieving method that seeks group members' agreement on the issue before making a final choice and, as a result, produces a very satisfactory answer for the group, is crucial in this situation.

Thus, **the purpose of this study** is to provide a method for group recommendations, which utilizes automated consensus reaching models to provide agreed-upon suggestions.

1. LITERATURE REVIEW

Group decision-making is a common and regular action carried out in companies in the present day. Hence, it is important to address group decision-making concerns that are relevant to ensure optimal progress inside a business. These dilemmas may be described as instances in which members of a group must collaboratively choose a solution from a collection of possible options [8].

Various functional views exist about the process of group decision-making, including issue analysis, goal setting, item identification, and item assessment and selection. Problem analysis enables a group to examine the probable reasons of a problem and identify the underlying issues or the symptoms associated with the problem [9]. Goal setting enables a group to choose the resolution to an issue that requires collective decision-making. Item identification facilitates the process of identifying potential solutions and encourages collective brainstorming within the group. Ultimately, the process of item assessment and selection empowers group members to assess the goods and choose the most superior option [10].

This literature review specifically examines the viewpoint in which group members use a preference structure to articulate their thoughts over a list of prospective things. Subsequently, a two-phase selection procedure is conducted to arrive at a definitive answer for a group decision-making issue.

In the first stage, the preferences of group members are collected and combined using aggregation algorithms. During the second phase, known as exploitation, a specific criterion is used to acquire an item or a collection of objects that will serve as the ultimate solution [11].

In a formal sense, a group decision-making issue is comprised of the following key components [12]:

– a collection U consisting of n users (group members), $U = \{u_1, \dots, u_n\}, n \geq 2$, which express their preferences for a collection of things;

– a collection X of m elements $X = \{x_1, \dots, x_m\}, m \geq 2$ to be selected as prospective answers to the challenge of making decisions as a group;

– set P represents the preferences of users about the things, which serves to express the opinions of users on the objects. The preference values are defined inside a rating domain $D(P \subseteq U \times X \to D)$.

A user's choice for an item may be denoted by a preference structure. Various preference structures have been used in group decision-making methodologies, including preference orderings, utility values, and preference relations [13]:

– preference ordering: A user, referred to as u_i expresses their preferences for a collection of m objects by creating an individual preference ordering. O^i is a set of permutations, denoted as $\{o_1^i, \ldots, o_m^i\}$, where $o^i(\cdot)$ represents a permutation function over the set of indexes $\{1, \ldots, m\}$. The user provides a ranked list of item choices in descending order. Within the context of recommendations, this preference structure may be described as “a hierarchical relationship between two or more items, used to determine which, out of a range of options, most closely aligns with the user's preferences”;

– utility values: a user, referred to as u_i, expresses their preferences for a collection of objects, denoted as X, using a set of m utility values. The set U_i consists of elements $u_1^i, \ldots, u_m^i$, where each element u_j^i belongs to the interval $[0, 1]$. The fundamental concept is that as the utility value of an item increases, so does the user's preference for the item's aims. Utility-based recommender systems, in which recommendations are generated by calculating the efficacy of each item for the user, have implemented this preference structure.

Several utility-elicitation techniques have been devised using Multi-Attribute Utility Theory to accurately capture a decision maker's comprehensive preferences;

– hierarchical rankings: the user's preferences, as defined by the function $\mu_p i: X \times X \to D$, explain the degree or intensity of preference for item x_j over item x_k in the domain D. This preference structure demonstrates the notion of paired preferences in recommendation situations. In this structure, instead of rating things individually, the user expresses their preferences by indicating which item they prefer in a pair (x_j, x_k). Users often naturally communicate their preferences in pairwise form in many real-life decision-making situations. When choosing a pair of shoes, we do not evaluate each pair of shoes independently. Alternatively, we are inclined to evaluate and then choose the desirable option [14].

Various forms of preference relations may be used based on the specific domain in which the strength of the preference is assessed. Out all these categories, fuzzy preference relations are the most used strategy because they are useful in simulating decision-making processes [15].

In this technique, if D is a value between 0 and 1, each value p_{jk}^i in the matrix P^i indicates the preference degree (related to user u_i) for item x_j over item x_k (typically, it is considered that $p_{jk}^i + p_{kj}^i = 1$, for all j and k):

– the equation $p_{jk}^i = 1/2$ implies that there are no discernible distinctions in the preferences of user u_i between items x_j and x_k;

– the equation $p_{jk}^i = 1$ signifies that item x_j is unequivocally favored over item x_k;

– the equation $p_{jk}^i = 0$ signifies that item x_k is unequivocally preferred over item x_j;

– the expression $p_{jk}^i > 1/2$ signifies that item x_j is chosen above item x_k. Another often used method is linguistic preference relations, which use a linguistic term, set to indicate the level of liking for the items. If D is equal to S, where S is a linguistic term set consisting of $s_0, \ldots, s_g$ with an odd number of elements $(g + 1)$, and $s_g/2$ represents a neutral label (such as “equally preferred”), while the labels p_{jk}^i in the matrix P^i denote the linguistic preference intensity of x_j over x_k [16].

In the context of group decision-making, the process of selecting a conclusion does not ensure a high degree of agreement in the outcome, which is crucial in several real-world scenarios. To address this limitation, it is necessary to implement a consensus-building procedure that may adjust the original preferences of individuals inside a group discussion, bringing them closer to a collective view that is agreeable to all members of the group [17].

Consensus is a condition in which all members of a group reach a mutual agreement, and the final choice is satisfactory to everyone. Consensus measurements quantify the degree of agreement among members of a group.

The measurements are bounded within the range $[0, 1]$, with 0 indicating no consensus and 1 indicating complete agreement.

The remaining assessment scores are within the range of $(0, 1)$, indicating degrees of partial unanimity. The consensus notion may be viewed from several perspectives, ranging from rigorous to more lenient interpretations, based on these principles [18].

Strict consensus, also known as unanimity, refers to a complete agreement among all members of a group. This means that the consensus measure is equal to 1. However, achieving strict consensus is sometimes difficult and expensive. The notion of soft consensus has been introduced as a means to relax the rigorous consensus approach, by defining consensus using fuzzy linguistic quantifiers.

2. PROPOSED METHOD

The overall structure of the proposed method is made up of the stages that are as follows:

– Recommendation – individual suggestions are generated for each member using a collaborative filtering mechanism during the first phase of the process, which is the recommendation phase. After that, these individual suggestions are filtered in order to identify the top N common groups of things. These sets of items are then represented as preference orderings, which will be used in the subsequent component;

– consensus – the individual suggestions are included into an automated consensus reaching process until a certain degree of agreement is attained. When all is said and done, the group suggestion and the collective preference are both computed.

2.1. Recommendation step

During this stage, the initial step is to compute the specific suggestions for each particular member.

In the case of every $u_j \in U$, the output is a collection of pairs t_k; $\tilde{r}_{u_j t_k}$ for every $t_k \in T - T_{u_j}$. These suggested goods are arranged in decreasing order after being classified according to rating prediction. Through this sorting, an ordering O_{u_j} is established for each individual member u_j.

As a result of the fact that customers have given varying ratings to various things, their orders will comprise distinct groupings of items. In order to rectify this situation, the ordering O_{u_j} takes into account just the things that are generally suggested. In other words, only the items that are included in set $T - M^G_{u_j} T_{u_j}$ are taken into consideration.

In the last step, the things that are ranked highest are chosen to go on to the subsequent phase. The collection of all ratings that are included inside the recommender system serves as the system's input. It is essential to take note of the fact that only a limited subset of items from the total number of available items are known, which is denoted by the symbol $R \subset T \times M$.

The application of single user collaborative filtering is the initial phase in the process. The first step is to create all of the individual forecasts for each $u_j \in U$ by utilizing their ratings over the items that are stored in the group recommendation system database respectively. These individual forecasts are created for things that the member has not previously evaluated; in other words, all of the predictions $\tilde{r}_{u_j t_k}$ are generated for $t_k \in T - T_{u_j}$.

Immediately after this, it is necessary to extract the common set of objects that are suggested. This step is carried out since the single user recommender system may not be able to give predictions for all $\langle u_j, t_k \rangle$ pairings, which is not acceptable for the subsequent consensus phase.

We are going to refer to the collection of things that are often expected for the group U as $T_{\tilde{R}_U}$:

$$T_{\tilde{R}_U} = \left\{ t_k \ s.t. \ \forall u_j \in U \ \exists \tilde{r}_{u_j t_k} \right\}. \tag{1}$$

Following the computation of the set of commonly anticipated things, the predictions are pre-filtered in order to decrease the number of items that will be part of the item set that will be used during the consensus phase. This is done to limit the amount of computing work required during the consensus phase and to shorten the list of things that are proposed.

Because of this selection, things that are not suitable for the group suggestion are eliminated from consideration. There are a variety of social choices voting systems that may be used for the selection process. Some examples of these systems are the Borda count [19], cumulative voting [20], and single transferrable voting [21].

2.2. Consensus step

During the first step, the individual suggestions of each member with reference to the common collection of things that are suggested are produced. In the phase of reaching agreement, the objective is to get a high degree of consensus among the individual proposals in order to gain the collective choice. In this step, we investigate the application of two of the automated consensus support models that are currently available: (i) the minimal cost consensus model, and (ii) the automatic consensus support system model that is based on feedback.

2.2.1. Minimal cost consensus model

In this particular configuration of the consensus phase, the individual predictions that were acquired in the recommendation phase are subjected to the application of the minimal cost consensus model [22]. The individual forecasts of each participant are received during this phase. These predictions are regarded as individual preference values within the context of this model. The minimal cost consensus model modifies the preferences of users in order to arrive at a consensus as quickly as possible.

Following the establishment of a consensus, the model will obtain one preference value for each item. This value represents the group rating prediction that will be used to make recommendations. It is necessary to apply a distinct instance of the minimal cost consensus model to each individual item ik.

Under each and every circumstance, the least cost consensus model is driven by the following rules:

– $\tilde{r}_{u_j t_k}$ is the rating forecast for the target item, and member u_j preference is the membership preference;

– $c_{u_j} = 1$ is the cost that is incurred when altering the preferences of a member u_j;

– the weight of a member $u_j$is equal to $w_{u_j} = 1/|U|$;

– $\varepsilon = 0.2$ and $\alpha = 0.8$.

In light of this, the following is the solution to the minimal cost consensus model that was solved using linear programming:

$$\begin{cases} min \sum_{u_j \in U} c_{u_j} \left|\hat{r}_{u_j t_k} - \tilde{r}_{u_j t_k}\right|, \\ \left|\hat{r}_{u_j t_k} - \hat{r}_{G t_k}\right| \leq \varepsilon, \qquad \forall u_j \in U, \\ s.t.\ \hat{r}_{G t_k} = \sum_{u_j \in U} w_{u_j} \hat{r}_{u_j t_k}, \\ \sum_{u_j \in U} w_{u_j} \left|\hat{r}_{u_j t_k} - \hat{r}_{G t_k}\right| \leq (1-\alpha). \end{cases} \tag{2}$$

where $\hat{r}_{u_j t_k}$ represents the choice of member u_j over item t_k at the conclusion of the consensus process, and $\hat{r}_{G t_k}$ represents the collective preference over item t_k at the conclusion of the procedure.

Once this model has been solved, the collective preference is used as the target item t_k forecast for the group, which is as follows:

$$Pred(U, t_k) = \hat{r}_{G t_k}. \tag{3}$$

The consensus approach can only be applied to the top k items from the social choice ranking, which is something that should be mentioned. When it comes to the group prediction, the average value is used in this particular scenario for items that are not included in the $top - k$ set.

2.2.2. Feedback-driven automatic consensus support model

Based on input, an automatic consensus support system model is presented. In this particular variation, an automated consensus support system acts as a simulation of the interaction that takes place between the members of a group and a moderator who makes suggestions for modifications that bring the members' preferences closer together [23].

When a particular degree of agreement has been obtained, the preference of the group is taken into consideration when calculating the suggestion of the group.

The first thing that has to be done is to describe individual predictions as fuzzy preference connections so that the consensus model can better understand them. Because of this, the individual predictions are transformed into crisp orderings.

Every single ordering $\tilde{O}_{u_j}$ is represented by a fuzzy preference relation [24]. A transformation function is being used by P_{u_j}.

Authors of paper [25] presented a number of transformation functions in order to address the many different preference representations that are present in decision making issues that are subject to ambiguity.

More specifically, they devised a transformation function in order to construct the fuzzy preference ordering from the crisp preference ordering according to the following:

$$P_{u_j} = \left(p_{u_j}^{t_k t_l}\right)_{(n \times n)}, \tag{4}$$

$$p_{u_k}^{t_k t_l} = \frac{1}{2}\left(1 + \frac{\tilde{O}_{u_j}(t_l) - \tilde{O}_{u_j}(t_k)}{n-1}\right). \tag{5}$$

This is an illustration of the first phase in the process. The suggestion for member u_1 is denoted by the equation $\tilde{O}_{u_1} = \{t_3, t_2, t_4, t_1\}$.

This suggestion is stated in a preference ordering, where $\tilde{O}_{u_1}(t_3) = 1$ indicates that item t_4 has the greatest rating for member u_1 due to the fact that it had the highest prediction.

The following is the fuzzy preference relation that corresponds to it:

$$P_{u_1} = \begin{pmatrix} - & 0.33 & 0.67 & 0.17 \\ 0.67 & - & 0.83 & 0.33 \\ 0.33 & 0.17 & - & 0 \\ 0.83 & 0.67 & 1 & - \end{pmatrix}, \quad (6)$$

where the value of $p_{u_1}^{t_1 t_2}$ is determined in accordance with Equation (5) as follows:

$$p_{u_1}^{t_1 t_2} = \frac{1}{2}\left(1 + \frac{\tilde{O}_{u_j}(t_2) - \tilde{O}_{u_j}(t_1)}{4-1}\right) = \frac{1}{2}\left(1 + \frac{2-3}{3}\right) = 0.33. \quad (7)$$

To push these preferences closer to each other gradually until the consensus level achieves the needed value, the second stage of the consensus phase is to utilize the fuzzy preference relation P_{u_j} of all members and perform a consensus reaching process.

This is done in order to bring the preferences closer together. In this case, an automated consensus model is used, which allows for the preferences to be automatically updated in order to bring them closer to one another. This is accomplished by the utilization of a feedback system that offers preference modifications to people [26].

When it comes to the group's fuzzy preference relations, the consensus reaching process starts monitoring the level of agreement that exists inside the group. A similarity matrix, denoted by the equation $SM_{u_j u_k} = \left(sm_{u_j u_k}^{t_l, t_m}\right)_{n \times n}$, is generated for every pair of members that belong to the same group.

The $sm_{u_j u_k}^{t_l, t_m}$ matrix represents the degree of similarity between members u_j and u_k in terms of their evaluations of items t_k and t_m.

Following the acquisition of the similarity matrices, the consensus reaching process will proceed to construct the consensus matrix $CM_{u_j u_k} = (cm^{u_j u_k})_{n \times n}$ by means of pairwise aggregation.

Each pairwise aggregation is an aggregation operator that is applied to the similarity values. Some examples of aggregation operators are the arithmetic mean and the OWA operator [27].

Last but not least, the overall consensus degree $cr \in [0,1]$ is generated by aggregating the values of the consensus matrix CM. This occurs when the consensus matrix CM is calculated.

To assess whether or not the members of the group have attained a sufficient degree of agreement, the consensus reaching process continues to examine the amount of consensus that exists inside the group.

Comparing the cr with the $\mu \in [0,1]$, which is a value that is established a priori and indicates the minimal degree of agreement that is necessary, is the method that is used to do this.

In the event that the degree of consensus cr is equal to or higher than μ, it indicates that there is sufficient agreement among the preferences of the members, and the procedure is completed.

In any other case, the consensus reaching process will proceed to alter the preferences of the members. Furthermore, the parameter Max_Rounds places a restriction on the number of rounds of update that may be performed.

The execution of the consensus process is the fundamental element of the consensus reaching process method. The purpose of this section is to discover the members of the group whose fuzzy preference matrix P_{u_j} is the most far from consensus by evaluating all of the fuzzy preferences matrices.

The first step is to generate a collective preference P_c by combining the individual evaluations of each pair of things under consideration. $PP_{u_j} = \left(pp_{u_j}^{t_k t_l}\right)_{(n \times n)}$ is the formula that is used to construct a proximity matrix for each $u_j \in U$. The $pp_{u_j}^{t_k t_l}$ variable represents the degree to which the member's opinion is similar to the collective preference with respect to each pair of pieces of information [28].

Following that, the use of the proximity matrix PP_{m_j} is utilized in order to identify individuals whose preferences are not sufficiently near to the consensus. During this procedure, preferences $pp_{u_j}^{t_k t_l}$ that have values that are not in agreement with the majority are identified. These preferences are then automatically modified [29] in order to raise the degree of group consensus cr.

Immediately after the completion of the consensus process, the consensus reaching process will proceed with a fresh round and carry out the consensus measurement itself. After either the requisite degree of agreement has been obtained or the maximum number of rounds has been completed, the procedure is said to have reached its conclusion.

The suggestions that have been agreed upon are computed with the help of the collective preference P_c. Based on the fact that a consensus reaching process was used in order to calculate P_c, it is evident that there is a high degree of consensus among the members of the group.

In conclusion, an exploitation phase is carried out in order to choose the many options about the collective choice P_c.

According to paper [30], the non-dominance selection criterion is the one that is used for this particular selection. In order to identify the stringent preference relation for the group, this criteria computes a nondominance degree for each item by using P_c.

The formulation of this criterion is as follows:

$$\tilde{p}_c^{t_j t_k} = \begin{cases} p_c^{t_j t_k} - p_c^{t_j t_k} \; if \; p_c^{t_j t_k} > p_c^{t_k t_j} \\ 0 \; otherwise \end{cases}. \quad (8)$$

The non-dominance degree, denoted by $ND(t_k)$ is then determined for each t_k in the following manner:

$$ND(t_k) = 1 - \max_{t_k} \left\{ \tilde{p}_c^{t_j t_k} \right\}. \quad (9)$$

When it comes to the group suggestion, the ultimate ranking of each item t_k is determined by the non-dominance degree, denoted by $ND(t_k)$, of that item.

3. EXPERIMENTAL RESULTS

This section provides a description of the experiment that was carried out in order to assess both variations of the proposed method and compare them with earlier techniques, such as the average or the least misery [31].

The purpose of this experiment was to determine whether or whether the process of obtaining agreement enhances group recommendation.

Initially, the approaches that are being compared are delineated. Following that, the datasets and the techniques that are used to handle them are described in depth. In a later stage, the assessment criteria are specified.

In the end, the outcomes of both trials are presented and assessed via the process. In conclusion, a visual demonstration of the group agreement effect for the group recommender system using the automated consensus support system model that is based on feedback is shown in the form of an example.

3.1. Method variants comparison

The purpose of the experiments is to evaluate the consensus method for group recommendation in comparison to alternative methods that are centered on the delivery of group recommendations that are acceptable for all members of the group. The user-based collaborative filtering and the item-based collaborative filtering are the two single user recommender systems that are examined in the first experiment.

The minimal cost consensus model variation is evaluated for both of these systems. Due to the fact that this model is assessed using three different configurations and compared with two additional procedures, the total number of variants that are compared is five:

– MinCost top-10: As a means of determining the final aggregate rating value, the minimum cost consensus model is exclusively implemented on the ten items deemed most excellent based on the social choice voting system. The final items are assigned the mean value;

– MinCost top-50: The minimal cost consensus model is implemented on the fifty most popular items as determined by the social choice voting system;

– MinCost all: Since the minimal cost consensus model is implemented on the entire set of items that are commonly predicted, it does not take into account the outcomes of the social choice voting system;

– Mean: The aggregate evaluation is calculated by averaging the predictions made by each individual regarding the objective item;

– Minimum: The collective rating for the objective item is calculated as the minimum of all individual predictions.

In the second experiment, the automated consensus support system model that is based on feedback variant is assessed and contrasted with the recommendation aggregation group recommender system operator that uses the minimum as the aggregation operator [32].

The user-based collaborative filtering is the individual recommender mechanism that comes into play here. As a result of a number of adjustments, this recommender system has been enhanced [33].

To determine the degree of resemblance between the two groups, this experiment makes use of Pearson's correlation coefficient. A relevance factor is used in order to punish similarities that are not calculated with a sufficient number of co-rated items.

This is done in light of the fact that data sparsity might introduce bias into the similarity. Twenty is the value that is utilized for the particular relevance factor. For the purpose of aggregating the ratings of neighbors, the weighted sum is used during the rating prediction phase.

The same selection of top-n items that was used in the consensus-driven group recommender system is used here for the purpose of ensuring that the findings are comparable. For the purpose of isolating the influence of the consensus reaching process itself, the proposal makes use of the exact same individual regression functions.

Evaluations are performed on a number of different configurations for the consensus reaching process using a variety of consensus degree criteria μ. During the experiment, we demonstrate the outcomes for the consensus degree values expressed as $\mu = \{0.8, 0.85, 0.9\}$.

3.2. Used dataset

The MovieLens dataset was used for the experimental study. To be more specific, we make use of the ml-100k version, which is comprised of one hundred thousand ratings statements that were provided by 943 people across 1682 movies that fall inside the $\{1,2,3,4,5\}$ domain. The MovieLens dataset does not include any information about groups. In light of this, the method of group creation that is used is the random group formation, in which the number of members in the group is established to be five.

Through the use of hold-out validation and a test set that is 20 percent of the total, the dataset is divided into training and test sets. Many different executions of this split have been carried out in order to acquire findings that can be relied upon. For the purpose of group recommendation, the hold-out approach has been modified to choose the ratings in the test set only from the items that were evaluated by each group.

3.3. Methods measurements

In these studies, three assessment metrics that are commonly used are employed in order to evaluate the outcomes of the framework in terms of its capacity to recommend: (i) the area under the receiver operator characteristic curve, (ii) precision, and (iii) mean absolute error.

For the purpose of evaluating the outputs of classifiers in relation to a threshold, the area under the receiver operating characteristic curve (AUC) is used. In recommender systems, the size of the suggestion list is the threshold that is taken into consideration.

To be more exact, the area under the curve (AUC) measurements how the sensitivity and specificity respond as the threshold is increased. Several points are generated as a result of this rise, which are characterized by its specificity and sensitivity.

These points constitute a curve, the size of which is equal to the area under the curve (AUC) of the classifier. The outcomes of the group recommender systems are improved in proportion to the value of the variable.

Precision [34] is a metric that is used to ascertain the degree of accuracy that the suggestions provided by the recommender system possess.

In particular, it evaluates the proportion of things in the suggestion that are considered to be genuine positives. In a manner similar to that of the AUC, the outcomes of the group recommender systems are improved when its value is higher.

In order to ascertain the degree of accuracy of the rating prediction made by a group recommender system, the Mean Absolute Error (MAE) algorithm is used. Because it is a measurement of inaccuracy, the group recommender system is considered to be of higher quality when its value is lower.

3.4. Experiments on group recommender system with minimum cost

The performance that was produced by the methodologies that were compared is shown in 1 and Table 2, respectively, with relation to the AUC and MAE.

Table 1. **Evaluation results (AUC)**

RS	Method variants				
	MinCost top-10	MinCost top-50	MinCost all	Mean	Min
User-based	0.6431	0.6450	0.6429	0.6431	0.6249
Item-based	0.5560	0.5432	0.5442	0.5532	0.5492

Source: **compiled by the authors**

Min-Cost top-50 with user-based collaborative filtering produced the best performance in the case of AUC, while the MinCost top-10 strategy got the greatest performance for item-based collaborative filtering.

Both of these approaches were successful in achieving the highest performance. When it comes to MAE, the greatest performance with user-based collaborative filtering was accomplished by Mean variant, and when it came to item-based collaborative filtering, the best performance was accomplished by MinCost top-10. It is important to point out that the Minimum approach got much lower outcomes in terms of MAE when compared to the other strategies that were used.

Table 2. **Evaluation results (MAE)**

RS	Method variants				
	MinCost top-10	MinCost top-50	MinCost all	Mean	Min
User-based	0.7748	0.7750	0.7751	0.7741	0.8679
Item-based	0.7988	0.8139	0.8152	0.7993	0.9169

Source: **compiled by the authors**

It is also important to note that the MinCost top-10 and MinCost top-50 obtain superior results when compared to MinCost all.This indicates that the use of automated consensus methods on a limited set not only lowers the amount of computing resources required, but also enhances the overall performance of the system.

In general, consensus models are able to enhance group suggestions in the majority of instances. Furthermore, when it comes to MAE with user-based collaborative filtering, consensus models attain a performance that is comparable to that of the most effective strategy.

3.5. Experiments on group recommender system with feedback-driven automatic consensus support model

The group recommender system with the automated consensus support system model based on feedback was the subject of the second experiment. The findings of the approaches that were compared with reference to their AUC are shown in Table 3.

Table 3. **AUC for evaluated variants**

Variant	AUC
Baseline	0.5319
Consensus 0.80	0.5798
Consensus 0.85	0.5787
Consensus 0.90	0.5788

***Source:* compiled by the authors**

Considering that the baseline findings are improved by the three different configurations of Consensus, it can be concluded that the consensus reaching process is beneficial to the suggestion.

With regard to this particular dataset, the best results are achieved when the consensus degree is set at 0.8. The findings of the procedures that were compared with respect to their precision are shown under Fig. 1.

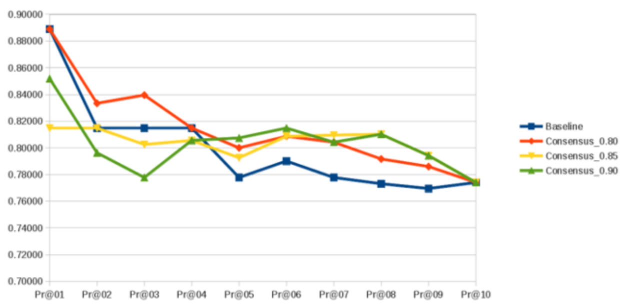

Fig. 1. **Precision for different list sizes for method variants**

***Source:* compiled by the authors**

The size of the list of recommendations is shown along the X axis, and the accuracy of such a list size is represented along the Y axis.

In terms of accuracy, the various procedures that are being compared provide different findings, which is proof that each methodology is giving distinct suggestions. This is something that can be seen. To be more specific, the proposal that has a consensus degree of 0.8 has the highest performance for a suggestion list that contains up to four items. The accuracy in the remaining situations is not the best when compared to the precision in the other configurations; nonetheless, it is superior to the findings of the baseline.

The results of Precision are computed for the same set of 10 items, which is something that should be brought to attention.

In light of this, the modification to the suggestion is the sorting that each method produces for the 10 items. Due to the fact that the precision does not take into account the order in which the things that are suggested are presented, it produces the same result whether the number of items on the recommendation list is 10.

Based on the accuracy value of 10 recommendations, it can be deduced that the test set had 77.5 % positive ratings.

CONCLUSIONS

In this paper, a method is presented for consensus-driven group recommender systems that incorporate consensus-building procedures into the recommendation procedure in order to increase member contentment with the recommendation. Two different iterations of the framework are provided. The initial one employs a minimum cost consensus model in order to enhance consensus while minimizing the necessary modifications. In order to enhance consensus and reach agreed-upon solutions, the second one simulates the negotiation process between experts and the moderator using a feedback-driven automated consensus support system model.

The performance of both models has been assessed and verified through experiments that compare various configurations of the proposed methods with those of the baseline techniques. The findings indicate that the incorporation of consensus-building processes into group recommender systems improves recommendation outcomes, with performance on the evaluated metrics surpassing that of the baseline.

REFERENCES

1. Hodovychenko, M. A. & Gorbatenko, A. A. “Recommender systems: models, challenges and opportunities”. *Herald of Advanced Information Technology*. 2023; 6 (4): 308–319. DOI: https://doi.org/10.15276/hait.06.2023.20.

2. Zhu, H., Ni, Y. & Zheng, Q. “A group-oriented recommendation algorithm based on similarities of personal learning generative networks”. *IEEE Access*. 2018; 1 (1): 42729–42739. DOI: https://doi.org/10.1109/ACCESS.2018.2856753.

3. Kunaver, M. “Diversity in recommender systems – A survey”. *Knowledge-Based Systems*. 2017; 123 (1): 154–162. DOI: https://doi.org/10.1016/j.knosys.2017.02.009.

4. Horpenko, D. R. “A conceptual model of decision-making support of the volunteer team in conditions of dynamic changes”. *Herald of Advanced Information Technology*. 2022; 5 (4): 275–286. DOI: https://doi.org/10.15276/hait.05.2022.20.

5. Kumar, C., Chowdary, C. R. & Shukla, D. “Automatically detecting groups using locality-sensitive hashing in group recommendations”. *Information Sciences*. 2022; 601 (1): 207–223. DOI: https://doi.org/10.1016/j.ins.2022.04.028.

6. Liu, T. “A review of deep learning-based recommender system in e-learning environments”. *Artificial Intelligence Review*. 2022; 55 (8): 5953–5980. DOI: https://doi.org/10.1007/s10462-022-10135-2.

7. Ziegler, C., McNee, S. & Lausen, G. “Improving recommendation lists through topic diversification”. *The 14th Internation Conference on World Wide Web*. 2005; 1 (1): 22–32. DOI: https://doi.org/10.1145/1060745.1060754.

8. Boratto, L. & Carta, S. “Art: group recommendation approaches for automatically detected groups”. *International Journal of Machine Learning and Cybernetics*. 2015; 6 (6): 953–980. DOI: https://doi.org/10.1007/s13042-015-0371-4.

9. Angulo, C. “Bridging cognitive models and recommender systems”. *Cognitive Computation*. 2020; 12 (2): 426–427. DOI: https://doi.org/10.1007/s12559-020-09719-3.

10. Fang, L. “Differentially private recommender system with variational autoencoders”. *Knowledge-Based Systems*. 2022; 250 (1): 109–130. DOI: https://doi.org/10.1016/j.knosys.2022.109044.

11. Cremonesi, P. & Turrin, R. “Performance of recommender algorithms on top-n recommendation tasks”. *The fourth ACM Conference on Recommender Systems*. 2010; 1 (1): 39–46. DOI: https://doi.org/10.1145/1864708.1864721.

12. Friedkin, N. & Johnsen, E. “A sociological examination of small group dynamics”. *Social Influence Network Theory*. 2011; 33 (1): 1–12. DOI: https://doi.org/10.1017/CBO9780511976735.

13. Koren, Y. & Volinsky, C. “Matrix factorization techniques for recommender systems”. *Computer*. 2009; 42 (8): 30–37. DOI: https://doi.org/10.1109/MC.2009.263.

14. Van Dat, N. “Solving distribution problems in content-based recommendation system with gaussian mixture model”. *Applied Intelligence*. 2022; 52 (2): 1602–1614. DOI: https://doi.org/10.1007/s10489-021-02429-9.

15. Zhang, S. “Deep learning-based recommender system: A survey and new perspectives”. *ACM computing surveys (CSUR)*. 2019; 52 (1): 1–38. DOI: https://doi.org/10.1145/3285029.

16. Huang, Z., & Zhou, M. “An efficient group recommendation model with multiattention-based neural networks”. *IEEE Transactions on Neural Networks and Learning Systems*. 2020; 31 (11): 4461–4474. DOI: https://doi.org/10.1109/tnnls.2019.2955567.

17. Serrano-Guerrero, J “A t1owa and aspect-based model for customizing recommendations on ecommerce”. *Applied Soft Computing*. 2020: 97 (1): 106–125. DOI: https://doi.org/10.1016/j.asoc.2020.106768.

18. Senot, C., Kostadinov, D. & Bernier, C. “Analysis of strategies for building group profiles”. *The 18th International Conference on User Modeling, Adaptation and Personalization*. 2010; 1 (1): 40–51. DOI: https://doi.org/10.1007/978-3-642-13470-8_6.

19. Deldjoo, Y. "Using visual features based on mpeg-7 and deep learning for movie recommendation". *International Journal of Multimedia Information Retrieval*. 2018; 7 (4): 207–219. DOI: https://doi.org/10.1007/s13735-018-0155-1.

20. Ray, B. "An ensemble-based hotel recommender system using sentiment analysis and aspect categorization of hotel reviews". *Applied Soft Computing*. 2021; 98 (1); 106–125. DOI: https://doi.org/10.1016/j.asoc.2020.106935.

21. Amato, F. "Sos: a multimedia recommender system for online social networks". *Future Generation Computer Systems*. 2019; 93 (1): 914–923. DOI: https://doi.org/10.1016/j.future.2017.04.028.

22. Wang, W., Zhang, G. & Lu, J. "Member contribution-based group recommender system". *Decision Support Systems*. 2016; 87 (1): 80 – 93. DOI: https://doi.org/10.1016/j.dss.2016.05.002.

23. Pham, P., Nguyen, L. T. T., Nguyen, N. T., Kozma, R. & Vo, B. "A hierarchical fused fuzzy deep neural network with heterogeneous network embedding for recommendation". *Information Sciences*. 2023; 620 (1): 105–124. DOI: https://doi.org/10.1016/j.ins.2022.11.085.

24. Schafer, J., Frankowski, D. & Sen, S. "Collaborative filtering recommender systems". *The Adaptive Web*. 2017; (1): 291–324. DOI: https://doi.org/10.1007/978-3-540-72079-9_9.

25. Guy, I. & Barnea, M. "Increasing activity in enterprise online communities using content recommendation". *ACM Transactions on Computer-Human Interaction*. 2016; 23 (4): 1–22. DOI: https://doi.org/10.1145/2910581.

26. Ma, X., Zhang, Y. & Zeng, J. "Newly published scientific papers recommendation in heterogeneous information networks". *Mobile Networks and Applications*. 2019; 24 (1): 69–79. DOI: https://doi.org/10.1007/s11036-018-1133-9.

27. Wu, C. "Knowledge graph-based multi-context-aware recommendation algorithm". *Information Sciences*. 2022; 595 (1): 179–194. DOI: https://doi.org/10.1016/j.ins.2022.02.054.

28. Huang, Z., Liu, Y., Zhan, C., Lin, C., Cai, W. & Chen, Y. "A novel group recommendation model with two-stage deep learning". *IEEE Transactions on Systems, Man, and Cybernetics: Systems*. 2021; 52 (9): 5853–5864. DOI: https://doi.org/10.1109/TSMC.2021.3131349.

29. Zhou, T., Kuscsik, Z. & Zhang, Y. "Solving the apparent diversity-accuracy dilemma of recommender systems". *Proceedings of the National Academy of Sciences of the United States of America*. 2010; 107 (10): 4511–4515. DOI: https://doi.org/10.1073/pnas.1000488107.

30. Cami, B. "User preferences modeling using dirichlet process mixture model for a content-based recommender system". *Knowledge-Based Systems*. 2019: 163 (1): 644–655. DOI: https://doi.org/10.1016/j.knosys.2018.09.028.

31. Ghanghas, Y., Rana, C. & Dhingra, S. "Diversity in recommender system". *International Journal of Engineering Trends and Technology*. 2013; 4 (1): 2344–2348. DOI: https://doi.org/10.1002/int.21922.

32. Masthoff, J. & Gatt, A. "In pursuit of satisfaction and the prevention of embarrassment: affective state in group recommender systems". *User Modelling and User-Adapted Interaction*. 2006; 16 (3): 281–319. DOI: https://doi.org/10.1007/s11257-006-9008-3.

33. Vijayakumar, V. "Effective knowledge-based recommender system for tailored multiple point of interest recommendation". *International Journal of Web Portals (IJWP)*. 2019; 11 (1): 1–18. DOI: https://doi.org/ 10.4018/IJWP.2019010101.

34. Gartrell, M., Xing, X., Lv, Q., Beach, A. & Seada, K. "Enhancing group recommendation by incorporating social relationship interactions". *The 16th ACM International Conference on Supporting Group Work*. 2010; 1 (1): 97–106. DOI: http://dx.doi.org/10.1145/1880071.1880087.

Conflicts of Interest: the authors declare no conflict of interest

Received 17.01.2024
Received after revision 14.03.2024
Accepted 21.03.2024

DOI: https://doi.org/10.15276/hait.07.2024.3
УДК 004.42

Досягнення консенсусу в групових рекомендаційних системах

Горбатенко Анастасія Артурівна[1)]
ORCID: https://orcid.org/0000-0002-5165-5168; nastya000511@gmail.com
Годовиченко Микола Анатолійович[1)]
ORCID: https://orcid.org/0000-0001-5422-3048; hodovychneko@op.edu.ua. Scopus Author ID: 57188700773
[1)] Національний університет «Одеська політехніка», пр. Шевченка, 1. Одеса, Україна, 65044

АНОТАЦІЯ

Традиційні системи групових рекомендацій не враховують вплив групової динаміки на групові рекомендації, наприклад, процес узгодження індивідуальних уподобань під час колективного прийняття рішень. Цей сценарій вже розглядався раніше в контексті групового прийняття рішень, зокрема, у зв'язку з процедурами досягнення консенсусу. У таких процесах експерти беруть участь у переговорах, щоб визначити свої уподобання і врешті-решт обрати взаємоузгоджений варіант. Мета процедури досягнення консенсусу - запобігти незадоволенню членів групи пропозицією. Попередні дослідження намагалися досягти цієї характеристики в групових рекомендаціях, використовуючи мінімальний оператор для процесу агрегування рекомендацій. Проте, використання цього оператора забезпечує лише мінімальний ступінь консенсусу щодо пропозиції, але не забезпечує задовільного рівня згоди між членами групи щодо групової рекомендації. Стаття присвячена аналізу процедур досягнення консенсусу в контексті групових рекомендацій для прийняття групових рішень. Метою дослідження є використання процесів досягнення консенсусу для надання групових рекомендацій, які задовольняють усіх членів групи. Крім того, дослідження спрямоване на вдосконалення систем групових рекомендацій шляхом забезпечення прийнятного рівня згоди між користувачами щодо групових рекомендацій. Таким чином, системи групових рекомендацій розширюються за рахунок включення механізмів досягнення консенсусу для полегшення прийняття групових рішень. У контексті групового прийняття рішень колективне рішення приймається групою осіб, які можуть бути фахівцями, з пулу варіантів або потенційних рішень проблеми, що розглядається. Для цього кожен фахівець отримує свої переваги щодо кожної можливості. Традиційні методи вибору для групового прийняття рішень не враховують можливості розбіжностей між експертами щодо обраного варіанту. Ця проблема вирішується шляхом використання методів досягнення консенсусу, коли досягається значний ступінь згоди перед вибором остаточного рішення. Щоб полегшити узгодження смаків експертів, вони неодноразово модифікують їх, щоб збільшити їхню близькість. Перед тим, як зробити колективний вибір, іноді необхідно досягти певного ступеня консенсусу. Таким чином, ця стаття представляє архітектуру групових рекомендацій, яка використовує автоматизовані моделі досягнення консенсусу для надання прийнятих пропозицій. Зокрема, ми розглядаємо модель консенсусу з мінімальною вартістю та модель автоматизованої системи підтримки консенсусу, яка покладається на вхідні дані. Модель консенсусу з мінімальною вартістю обчислює колективну пропозицію групи шляхом коригування індивідуальних переваг на основі функції вартості. Це досягається за допомогою лінійного програмування. Модель автоматизованої системи підтримки консенсусу на основі зворотного зв'язку імітує взаємодію між членами групи та модератором. Модератор пропонує коригування індивідуальних пропозицій, щоб зблизити їх і досягти високого ступеня узгодженості перед тим, як сформувати групову рекомендацію. Обидві моделі оцінюються і порівнюються з базовими процедурами під час тестування.

Ключові слова: Рекомендаційна система; машинне навчання; нейронні мережі; глибоке навчання; класифікація; система фільтрації інформації; інформаційна система

ABOUT THE AUTHORS

Anastasiia A. Gorbatenko - PhD Student of Information Systems Department. Odessa Polytechnic National University, 1, Shevchenko Ave. Odessa, 65044, Ukraine
ORCID: https://orcid.org/0000-0002-5165-5168; nastya000511@gmail.com
***Research field*:** Deep learning; data mining; smart cities; video processing; motion tracking; project-based learning; patter recognition

Горбатенко Анастасія Артурівна - аспірант кафедри Інформаційних систем. Національний університет «Одеська Політехніка», пр. Шевченка, 1. Одеса, 65044, Україна

Mykola A. Hodovychenko - PhD, Associate professor of the Artificial Intelligence and Data Analysis Department, Odessa Polytechnic National University, 1, Shevchenko Av. Odessa, 65044, Ukraine
ORCID: https://orcid.org/0000-0001-5422-3048; hodovychenko@op.edu.ua. Scopus Author ID: 57188700773
***Research field*:** Deep learning; data mining; smart cities; video processing; motion tracking; project-based learning; patter recognition

Годовиченко Микола Анатолійович - кандидат технічних наук, доцент кафедри Штучного інтелекту та аналізу даних. Національний університет «Одеська політехніка», пр. Шевченка, 1. Одеса, 65044, Україна