Decision Support Systems 54 (2013) 1055–1064

Contents lists available at SciVerse ScienceDirect

Decision Support Systems

journal homepage: www.elsevier.com/locate/dss

Mining consensus preference graphs from users' ranking data

Yen-Liang Chen [a,*], Li-Chen Cheng [b], Po-Hsiang Huang [a]

[a] Department of Information Management, National Central University, Chung-Li 320, Taiwan, ROC
[b] Department of Computer Science and Information Management, Soochow University, Taipei 100, Taiwan, ROC

ARTICLE INFO

Article history:
Received 1 October 2011
Received in revised form 28 September 2012
Accepted 21 October 2012
Available online 27 October 2012

Keywords:
Data mining
Decision making
Group decision making
Maximum consensus sequence
Preference graph

ABSTRACT

The group ranking problem consists of constructing coherent aggregated results from preference data provided by decision makers. Traditionally, the output of a group ranking problem can be classified into ranking lists and maximum consensus sequences. In this study, we propose a consensus preference graph approach to represent the coherent aggregated results of users' preferences. The advantages of our approach are that (1) the graph is built based on users' consensuses, (2) the graph can be understood intuitively, and (3) the relationships between items can be easily seen. An algorithm is developed to construct the consensus preference graph from users' total ranking data. Finally, extensive experiments are carried out using synthetic and real data sets. The experimental results indicate that the proposed method is computationally efficient, and can effectively identify consensus graphs.

1. Introduction

The group ranking decision process involves aggregating individual rankings to obtain a representative group ranking. In other words, the group ranking algorithm generates consolidated ranking results that represent the group will, preference, or decision based on decision makers' preference data. In recent decades, the group ranking problem has become an important and interesting issue in decision making [11,15], machine learning [14], web search strategies [2,7] and others. The essence of this problem is how to consolidate and aggregate decision makers' rankings to obtain a group ranking that represents "better coherent" ordering in regards to the decision makers' rankings.

Generally, the traditional group ranking problem can be classified using three aspects: the completeness of the user-provided preference information, the input format used to express users' preferences, and the type of compromised output results. The group ranking problem can be roughly classified into two major approaches based on the completeness of the decision maker's preference information: the total ranking approach [7,17,20–23] and the partial ranking approach [3,4,8–10,18,19]. The former requires individuals to appraise all items (called alternatives), while the latter appraises only a subset of items. There are three typical input formats decision makers can use express their item preferences: weighting models, pair-wise comparisons, and ranking lists. All three formats have been used in previous studies to express individuals' input preferences. These formats may not be perfect, but they express user preferences reasonably well in most practical situations. Depending on the input format adopted, users are asked to rank (in the ranking list model), rate (in the weighting model), or compare (in the pair-wise comparison model) the items. After all preference data have been collected, an algorithm is applied to generate the consolidated output results. In previous research, output results could be divided into two main types. One is a total ranking list, which is an ordering list of all items that represent the achieved consensus. The other is a maximum consensus sequence, which gives the longest ranking lists of items that agree with the majority and disagree with the minority.

Unfortunately, both output formats have their own weaknesses. Most previous ranking list approaches attempted to minimize the total disagreement between multiple input rankings in order to obtain an overall ranking list that represented the achieved consensus. This disregards the fact that user opinions may be discordant and have no consensus, forcing a complete ranking result even if there is no consensus or only a slight consensus. In such a situation, what we obtain is merely the algorithm output, since different algorithms derive different ranking results due to their different designs. To overcome this weakness, Chen and Cheng [5,6] proposed the maximum consensus approach, which generates only those maximum sequences on which users have consensus, meaning they are agreed upon by a majority of users and disagreed with by a minority of users. However, this approach may generate many maximum consensus sequences, making the results fragmented and difficult to understand and use.

Therefore, we propose a method that finds consensus preferences and represents these relationships as a graph. This is called a preference graph, where the relationships are agreed upon by majority of users and disagreed with by only a minority of users. Accordingly, we develop algorithms to discover preference graphs from users' ranking lists, and use the graphs to present the preferences of all users.

* Corresponding author. Tel.: +886 3 4267266; fax: +886 3 4254604.
E-mail address: ylchen@mgt.ncu.edu.tw (Y.-L. Chen).

0167-9236/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.dss.2012.10.031

Table 1
A sample database.

u_{id}	User sequence
S1	$S_1=\{A=C>B=D>E\}$
S2	$S_2=\{A>C=D\geq B>E\}$
S3	$S_3=\{C>A=B\geq D>E\}$

Table 2
Dimensions of group ranking problem.

Completeness of preference information	Total ranking	Partial ranking	
Input format	Weighting models	Pair-wise comparisons	Ranking lists
Output format	Ranking orders	Consensus lists	Preference graph

Example 1. Suppose we have the three ranking lists shown in Table 1. We will show their consolidated results in the ranked order, maximum consensus sequence, and preference graph formats.

Using a total ranking list, we may get the result $\{A\geq C>B\geq D>E\}$, which represents a coherent ranking of all items. There is no consensus, however, on the rankings of A and C in the preference data. The reason they are arranged that way is simply because we are forced into a complete ranking.

Using maximum consensus sequences, the longest patterns of coherent item rankings are $\{A\geq D>E\}$ and $\{C>B>E\}$. The problem is that it may output many consensus sequences that need to be checked. Additionally, the preference between items A and C is unknown.

Using our approach, the result may look like Fig. 1. Items in the same cluster are similarly preferred by users. Therefore, items A and C are similarly preferred, and B and D are similarly preferred. Additionally, G_1 is preferred more than G_2 and G_3, and G_2 is preferred more than G_3. There are several advantages to this approach. First, the graph is built based on users' consensuses. Second, the graph can be intuitively understood. Third, relationships between items can be observed from the graph.

This paper is divided into six sections. Our motivations are discussed in Section 1. Section 2 reviews related works. Section 3 defines the problem of mining consensus preference graphs and provides definitions. Section 4 introduces the preference graph mining algorithm. Experimental results are presented in Section 5. Finally, we draw conclusions in Section 6.

2. Related work

In this section, we review literature regarding the group ranking problem. As shown in Table 2, the group ranking problem can be classified using three features: the completeness of input preference information, the type of input format, and the compromised output format.

When looking at the completeness of users' item appraisals, the group ranking problem can be identified as using the total ranking approach or the partial ranking approach. In the total ranking approach, all individuals have to appraise all items, no matter the input format. Originally, Kemeny and Snell studied the group ranking problem in a pair-wise format by minimizing the distance between two complete rankings [20]. This optimal approach has been proven as an NP-hard problem [1]. Following that, Cohen [7] developed a greedy-like algorithm that minimizes the discrepancy in rankings by integrating preference data. Additionally, Saaty developed the AHP (Analytical hierarchical process) method, a leading approach to multi-criteria decision making that consolidates many full ranking lists by assigning weights to different criteria [14,22,23].

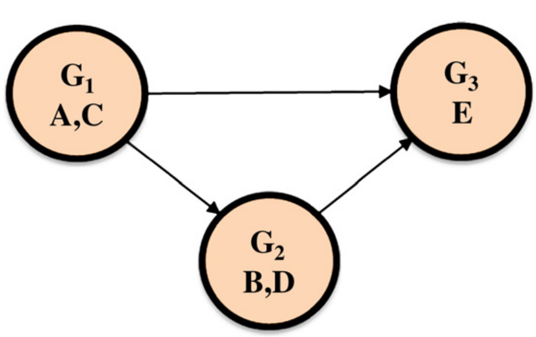

Fig. 1. Preference graph.

In the partial ranking approach, users rank a selected subset of items to avoid appraising unfamiliar items. Many models have been proposed to deal with partial rankings. For instance, Bogart extended the Kemeny–Snell model by allowing users to appraise a subset of items, and then finish a partial ordering [3,4]. Various solution methods based on distance functions have also been studied [8–10]. In some group ranking problems, rankings include intensity [19]. The goal of most partial ranking approaches is to obtain a full ranking from user input data. Cook proved that in some cases, however, the aggregated results will not be a full order [11].

Individual preference data can be expressed as item weights/scores, sets of pair-wise comparisons, or item ranking lists. Actually, scoring behavior varies from person to person [19]. It is difficult for users to accurately express his/her preferences as a precise numerical value [13]. Pair-wise comparisons are widely used, but create a large amount of work due to numerous comparisons [8,12,19]. If there are many items, comparing them all may become tedious. Finally, ranking lists can be total ordering or partial ordering lists. Compared to total ordering, partial ordering lists are generally easier to use, but tend to result in incomplete and inconsistent information problems.

Typically, the group ranking problem has two output formats: a total ranking list or maximum consensus sequences. A total ranking list is a full ordering list that has found a consensus with minimum disagreement from all preference data. From this list, we can easily see the relationships between items. As mentioned in the introduction, however, even when there is no consensus, we are forced into a total ranking list. As a result, what we obtain is merely the algorithm output. Therefore, maximum consensus sequences were proposed to solve this problem [5,6]. These sequences represent the consensuses that are agreed upon by a majority of users and disagreed with by a minority of users. However, since we may generate numerous maximum consensuses of different lengths, the results are fragmented and difficult to understand and use.

In our work, as shown in Table 2, we use the total ranking approach in regards to information completeness, ranking lists as the input, and preference graphs as the output format. Preference graphs are a brand new output format that has not been used in any previous research.

3. Problem definition

In this section, we formally define the problem of mining consensus preference graphs from users' ranking data. Let $U=\{u_1, u_2,\ldots,u_m\}$ and $I=\{i_1, i_2,\ldots,i_n\}$ denote the sets of all users and all items, respectively. Each user u_i creates a ranked list of all items that expresses his/her preferences. The ranked list of user u_i can be represented as a sequence $S_i=\{a_1\oplus a_2\oplus\ldots\oplus a_n\}$.

Each user sequence must satisfy the following conditions: first, an item $a_j\in I$, where $1\leq j\leq n$, cannot appear more than once in a user sequence; second, the comparator $\oplus$ belongs to $\{>, \geq, =\}$. The comparator "$>$" means that the preceding item is more preferable than the succeeding item. The comparator "$\geq$" indicates that the preceding item is at least as preferable as the succeeding item. Finally, the comparator "$=$" denotes the same preference for both items. For example, in