Improvement of similarity-diversity trade-off in recommender systems based on a facility location model

ANTIOPI PANTELI[*§] and BASILIS BOUTSINAS[†§]

[§] *Dept. of Business Administration, MIS & Business Intelligence Lab, University of Patras, Rio Patras, 26500, Greece*

There is a growing interest in the offering of novel alternative choices to users of recommender systems. These recommendations should match the target query while at the same time they should be diverse with each other in order to provide useful alternatives to the user, i.e. novel recommendations. In this paper, the problem of extracting novel recommendations, under the similarity-diversity trade-off, is modeled as a facility location problem. We formulate this trade-off as a multiple *p*-median problem solved by using biclustering. The results from tests in the benchmark Travel Case Base were satisfactory when compared to well-known recommender techniques, in terms of both similarity and diversity. Moreover, the experimental tests have shown that the proposed method is flexible enough, since a parameter of the adopted facility location model constitutes a regulator for the trade-off between similarity and diversity.

Keywords: Recommender Systems; similarity-diversity trade-off; facility location model; biclustering.

1. Introduction

The development of the Internet has resulted in an overload of data and users often find it difficult to extract information that best corresponds to their preferences or needs. Thus, recommender systems became part of life since they can manage and process the available information in order to filter the redundant part and extract useful knowledge (Perugini *et al.*, 2004). Recommender systems aim to reduce complexity in human life through selecting from a very large amount of information the part that is relevant to the active user (Lorenzi *et al.*, 2005). Thus, their applications can be found in different aspects of everyday life such as health (Trang Tran *et al.*, 2017), music (Chen & Chen, 2005), movies (Golbeck, 2006), travel (Fesenmaier *et al.*, 2003; Shih *et al.*, 2011), news (H. J. Lee & Park, 2007), e-learning (Bobadilla *et al.*, 2009) etc.

Many Web sites use recommender systems in order to efficiently exploit all the information recorded in their case bases. There are some major types of recommender systems that have been studied in the literature, such as the content-based systems (CBS), the collaborative filtering approaches (CF), the knowledge-based (KBS) systems, as well as hybrid approaches.

[*] Corresponding author. Email: antiopi.panteli@gmail.com
[†] Email: vutsinas@upatras.gr

In the first category (CBS), the system recommends items that are similar to the ones that the user liked in the past, while in the second (CF) the system recommends to the user the items that other users with similar tastes liked in the past. The similarity in preferences between two users is counted on the basis of their rating history. Therefore, in a collaborative approach, the system recommends items that are not in the active user's profile but that have been rated highly by other users who have similar preferences.

The knowledge-based systems (KBS) take into account the features of the items and examine how they meet users' needs and preferences. Thus, these systems recommend items by considering how useful they are in general for the user. In CF approaches, user ratings are necessary but the item descriptions (usually attribute-value pairs) are not required, whereas in KB approaches the retrieval set contains items whose descriptions match the user's query.

Case-based systems that are studied in this paper constitute a subclass of KBS. More specifically, the primary components of Case-based systems are:

1. a data base of previously solved problems along with their solutions
2. the user query which specifies the needs and preferences of the user in the form of attribute-value pairs
3. a similarity function which estimates how the user's query matches the cases of the case base and thus, the cases of the retrieval set (Ricci *et al.*, 2011).

Although recommender systems have developed significantly over the years, they still constitute an attractive research area due to many practical applications. Case-based Recommender Systems have been used in the literature in order to provide recommendations concerning different aspects of everyday life. For instance, case-based recommender systems have been used in heart disease diagnosis (Prakash, 2015) and wealth management services aiming to achieve personalized finance advisory, where investment proposals are adapted on the ground of a case base of previously proposed investments (Musto & Semeraro, 2015). Another application refers to the Massive Open Online Courses (MOOCs), where case-based recommender systems try to address the problem faced by various learners in finding the appropriate online courses that best match their personal interests. Thus, systems try to fit the user request based on her profile, needs and knowledge (Bousbahi & Chorfi, 2015). In general, it is common that the recommendation systems are based on the predefined preferences of the users and their purchase history (Chen & Chen, 2005). Also, in J. S. Lee & Lee (2007), a music recommendation system takes into account not only these factors, but also the user's situation or in other words the user's context at the time of making recommendation.

Recommender systems try to reduce the computational cost of managing, processing and searching huge amounts of data. For this reason, incremental algorithms are designed. When the volume of data increases, these algorithms perform incremental rather than global computations by slightly modifying already proposed results according to the newly added data (Sarwar *et al.*, 2002).

Recommender systems still face many challenges and require improvements in terms of personalized recommendations, results' accuracy, diverse recommendation lists, etc. New users that enter the system have inserted little or no information at all, which increases the difficulty of producing recommendations. Moreover, due to the large extent of available options, it is often difficult to identify and deliver the right content to the right person, which implies that personalized recommendation is required (Tiihonen & Felfernig, 2017). As a result, the algorithms that perform the recommendations should be able to provide different objects to different users, thus guaranteeing the inter-diversity of recommendations.

The recommendations have to be quite similar to the inserted query, but also they have to be different to each other in order to provide diverse options to the user (Hurley & Zhang, 2011; Smyth & McClave, 2001; Zhou *et al.*, 2010; Ziegler *et al.*, 2005), since users may get bored after receiving many recommended objects under the same topic. This type of diversity is called intra-user diversity. Bradley & Smyth (2001) described the diversity in the recommendation procedure as the opposite of similarity. In fact, the diversity of the recommendation list for a single user as stated before, which refers to the different objects that are proposed, provides significant value to the user since it offers solution to the over-fitting problem, i.e. recommends objects which might not be proposed otherwise, since they are not predictable by the known data. It has been shown that the diversity of proposed results increases the level of satisfaction of the target user. This derives from the fact that different objects which match the user's requirements and preferences offer a great range of options. However, the most common problem in the study of recommender systems, is the trade-off between similarity and diversity.

In this work, we apply a facility location model to improve the similarity-diversity trade-off in recommender systems. To the best of our knowledge, the application of the used location model to recommender systems has never been studied in the literature before. In that sense, our work can broaden the perspectives of the interaction and combination of different scientific fields in order to achieve the best possible results. We evaluate the proposed approach by using a benchmark recommender system that offers alternative choices in order to propose diverse travel recommendation plans to users, by selecting a specific number of cases from the Case Base that match the user's requirements and preferences expressed in the target query.

An important characteristic of the proposed approach is that a specific parameter of the facility location model constitutes a regulator for the trade-off between similarity and diversity of the recommendation set.

2. Related Work

The recommendation algorithms are evaluated in terms of accuracy through measures like the ranking score, the mean absolute error, the hitting rate, precision, recall, etc. (Zhou *et al.*, 2009). However, in the paper of Herlocker *et al.* (2004), it has been stated that accuracy is not enough and user satisfaction in terms of the recommendation list depends on other factors as well. This derives probably from the fact that when the accuracy and similarity of the recommended objects to a given query are high, it is possible that the objects are similar with each other as well and thus diverse choices are not really offered to the user. Therefore, the recommendation list should contain items that are similar to the user inserted query, but are also significantly diverse to each other at the same time. In a survey about query result diversification it is stated that diversity contributes to less simplistic results and can bring new information not previously mentioned (Zheng *et al.*, 2017). Also, since the users' queries are sometimes ambiguous, diversity can be used in order to provide results with varying information that may satisfy the users' true intentions. Thus, result diversification can be considered as an effective solution to minimization of query abandonment (Agrawal *et al.*, 2009; Clarke *et al.*, 2008; Das Sarma *et al.*, 2008).

Most works in the literature related to diversity enhancement in recommendation lists refer to collaborative filtering approaches rather than to case-based approaches. Such approaches can be found in the papers of Abbassi *et al.* (2013), Adomavicius & Kwon (2012), Aytekin & Karakaya (2014), Boim *et al.* (2011), Choi & Han (2010), Zhang & Hurley (2008) and Ziegler *et al.* (2005). The common problem in recommendation strategies is the trade-off between similarity and diversity. In Zhang & Hurley (2008) the competing objectives of diversity maximization in the retrieved recommendation set and the maintenance of adequate levels of similarity to the user query were modeled as a binary optimization problem. In Premchaiswadi *et al.* (2013) a ranking method was proposed that improves the overall recommendation diversity by taking into account the diversity impact of each item on the final recommendation set. In Choi & Han (2010) more diverse recommendations were offered through the calculation of category correlations and in Abbassi *et al.* (2013) the diversity was increased by avoiding showing items of the same category. In Ziegler *et al.* (2005) the authors computed the intra-list similarity which is a decreasing rather than increasing version of the function of diversity that was used in Smyth & McClave (2001). They provided a heuristic method to increase diversity and their results showed that the user's satisfaction goes beyond accuracy and takes into consideration other factors as well and mostly the diversification of the recommendation list. D. M. Fleder & Hosanagar (2007) considered the effect of recommender systems on the diversity of sales and used Gini coefficient to measure sales diversity.

The trade-off between diversity and similarity for case-based recommender systems has been studied much less than for collaborative filtering approaches. This is one of the reasons that motivated our research in this topic. However, it should be mentioned that there are some research works related to this trade-off in case-based recommender systems (Bradley & Smyth (2001); Hurley & Zhang, 2011; McSherry, 2002; Smyth & McClave, 2001).

More specifically, in Bradley & Smyth (2001) and Smyth & McClave (2001), a greedy selection algorithm is proposed to address the problem of diversity in case-based recommender systems. This method first sorts the items according to their similarity to the target query and then builds incrementally the recommendation list in a way that tries to optimize both similarity and diversity. In the first iteration, the most similar item to the target query is selected and then in each iteration, the item with the maximum combination of similarity to the target query and diversity to the already selected items within the recommendation list is selected. The algorithm continues until the desired size of the recommendation list is achieved. However, this method is inefficient and in Smyth & McClave (2001), a bounded version of the algorithm was proposed. This improved version of the algorithm first selects a certain number, say *b*, of items that are the most similar to the target query and then applies the initial greedy selection algorithm to these *b* items rather than applying it to the entire initial set of items. However, as *b* approaches the size of the original set of items, the complexity of this version of the algorithm approaches the complexity of the greedy selection method.

Another more recent algorithm that addressed the problem of the trade-off between similarity and diversity was proposed in Hurley & Zhang (2011). They further developed the approach proposed in McSherry (2002) and represented the trade-off between similarity and diversity as a quadratic programming problem and a control parameter was introduced that determines the importance of the level of diversification within the recommendation list. The obtained results by this method were quite similar to the ones of the bounded greedy selection method of Smyth & McClave (2001) but with slightly worse computational time complexity. Also, they proposed a new precision-based evaluation methodology by considering the item novelty and the ability of the recommendation strategy to suggest novel but also relevant items to the user preferences. Novelty refers to how different the recommended objects are with respect to what the users have already seen before.

Furthermore, it should be mentioned that in Hurley & Zhang (2011) the methods that were demonstrated can be applied both to collaborative filtering recommendation and case-based recommendation.

The majority of studies regarding new recommendation methods increase the diversity of the retrieval set for a given *individual* user, measured by the average dissimilarity between all pairs of recommended items, while maintaining an adequate level of accuracy (Bradley & Smyth, 2001; McSherry, 2002; Smyth & McClave, 2001; Zhang & Hurley, 2008; Ziegler *et al.*, 2005). However, in contrast to individual diversity, there are some studies (Adomavicius & Kwon, 2012; D. Fleder & Hosanagar, 2009) that refer to the aggregate diversity of recommendations across all users. In this paper, individual diversity is examined in the recommendation list.

3. Background

3.1. *Biclustering*

Biclustering techniques perform simultaneous clustering on rows and columns of the input data matrix. The term biclustering was first introduced by Cheng & Church

(2000) for the simultaneous clustering of gene expression data in DNA microarray analysis.

The biclustering technique has been used in various domains such as DNA microarray analysis, machine-part cell formation (Boutsinas, 2013a), electoral data analysis (Hartigan, 1972), text mining (Dhillon, 2001), nutritional data analysis (Lazzeroni & Owen, 2002), drug activity (Liu & Wang, 2003), agriculture (Mucherino & Urtubia, 2010), foreign exchange data (Lazzeroni & Owen, 2002) and target marketing (Ungar & Foster, 1998; Yang *et al.*, 2002). Moreover, biclustering is used to analyze operational purchase records in recommendation systems which use the collaborative filtering technique (Deodhar & Ghosh, 2007; Ungar & Foster, 1998) based on the idea that if users A and B share some identical preferences on certain products, then it is possible that the user A also likes other products that user B already likes (and vice-versa).

Most interesting proposed biclustering problems are proved to be NP-complete (Ben-Dor *et al.*, 2003; Cheng & Church, 2000; Tanay *et al.*, 2002) either searching for a minimum set of overlapping (or mutually exclusive) biclusters or searching for one "large" bicluster. As a result, most of the developed biclustering algorithms are based on heuristics (Busygin *et al.*, 2008; Madeira & Oliveira, 2004).

The adopted heuristic method for solving the multiple p-median facility location model (Panteli *et al.*, 2018) is based also on a heuristic biclustering algorithm (Boutsinas, 2013b). It is based on the idea of association rule mining in order to reduce the search space and enumerate the candidate biclusters. Of course, any of the methods that can solve effectively the multiple p-median facility location model can be also used instead.

The adopted heuristic biclustering algorithm (Boutsinas, 2013b) takes as input a Boolean data matrix. In a Boolean input data matrix $D(R, C)$, $R = \{r_1, r_2, \dots, r_o\}$, $C = \{c_1, c_2, \dots, c_a\}$, cell values $v_{ij} = 1$ $(v_{ij} = 0)$ denote that row r_i, $1 \leq i \leq o$, satisfies (does not satisfy) column c_j, $1 \leq j \leq a$. The algorithm is as follows:

(1) At first, the Boolean input data matrix is transformed to the matrix $D'(R, C')$, $C' = \{c'_1, c'_2, \dots, c'_b\}$, so that each row is assigned the column indexes it satisfies, i.e. $D'(r_i, c'_j) = \{c''_j | D(r_i, c''_j) = 1\}$

(2) Then, the transformed input matrix is processed by the Apriori Association Rule Mining algorithm.

(3) Let $fr(s)$ denote an extracted frequent itemset, where $fr \subseteq C$ and s is its support. Then, it is proved (Boutsinas, 2013b) that every frequent itemset $fr(s)$ defines a bicluster including a fraction s of the rows and $|fr|$ columns.

3.2. *The P-median Problem (PMP)*

In this subsection, we give a brief overview of a well-known facility location problem, the p-median problem, which was introduced by Hakimi (1964). It consists of locating *p* facilities that serve a given set of demand points such that the total distance between demand points and facilities is minimized.

Since the PMP is NP-hard (Kariv & Hakimi, 1979), exact methods can be used efficiently for smaller instances which contain several hundred nodes (demand points and facilities) (Gary & Johnson, 1979). However, for larger instances, e.g. including 1000 nodes, exact methods become inefficient, since computational time increases rapidly with instance size. As a result, heuristic approaches have been proposed in order to handle large sized problems by providing solutions slightly inferior to the optimal ones in much less computational time than the exact methods.

The first formulation of the problem as a zero-one programming problem can be found in the paper of ReVelle & Swain (1970). Some exact solution methods were presented in Beasley (1985), Hanjoul & Peeters (1985) and Rosenwein (1994). On the other hand, well-known heuristics can be found in Densham & Rushton (1992), Maranzana (1964) and Teitz & Bart (1968). A new heuristic for large-scale PMP instances was proposed in Avella *et al.* (2012). An overview of meta-heuristic approaches like tabu search, variable neighborhood search and simulation annealing that can escape from local optima and may effectively solve the p-median problem can be found in Mladenović *et al.* (2007). Genetic algorithms which are heuristic search methods that are designed to mimic the evolution process were also proposed in Bozkaya *et al.* (2002) and Alp *et al.* (2003). Another heuristic method is the well-known Variable Neighborhood Search Algorithm (VNS), which is based on the systematic change of neighborhood within a local search (Hansen & Mladenović, 1997).

The methods applied to the solution procedure of the p-median problem are many due to its wide range of applications in everyday life, such as in emergency facilities (fire stations, police control, ambulances) where the average response time of a facility should be minimized, in the location of production, storage and public facilities (Drezner & Hamacher, 2001), in telephone switching centers (Hakimi, 1964), in audit offices for a comptroller (Fitzsimmons & Allen, 1983), etc. The original formulation of the p-median problem or various extended formats are often incorporated into multi-objective models for locating facilities in the presence of multiple conflicting criteria (e.g. the models proposed in Mitropoulos *et al.* (2006) and Mitropoulos *et al.* (2013) for designing a network of primary health care providers).

Since uncertainty is inherent in most real life applications, there is often a possibility that each facility may fail and may not be able to satisfy the given demand due to weather, labor actions, electricity problems and other factors. The concept of facility failure is relatively new in the literature as it is different from approaches that refer to uncertain future conditions related to demands or costs. As a result, it is important to choose facility locations in a way that minimizes the cost while also hedging against failures, ensuring thus the supply-side robustness.

In Panteli *et al.* (2018), an extension of the original *p*-median problem, the multiple *p*-median problem (MPMP), was solved. This problem focuses on the possibility of service of a demand point more than one times, expressed by a parameter (mc). The MPMP can be applied in various situations where it may be necessary to provide backup facilities that can cover the demand in case the primary facility assigned to a demand point becomes unavailable.

The adopted heuristic algorithm for the MPMP takes as input a Boolean data matrix $D(R, C)$. Cell values of $D(R, C)$ represent the weight (e.g. distance) between specific demand points (represented by rows) and facilities (represented by columns). The algorithm is as follows:

(1) First, the set of biclusters B is extracted.

(2) Then, it searches in B to select the best bicluster. A bicluster is selected on the basis of two criteria, the coverage that each bicluster offers and its weight i.e. the distance between the columns of the bicluster (that represent the facilities) and the additional rows (that represent the demand points) that are covered when the bicluster is selected.

(3) Then, the previous step is repeated until the desired number of columns is selected.

In this paper, we broaden the perspectives of applications of the MPMP since the backup option does not represent only the notion of facility failure in covering the demand but also it represents the notion of multiple options. Thus, it represents the alternative choices which are recommended to the user by a recommender system. These alternatives have to match the target query's characteristics and at the same time have to be diverse with each other, in order to constitute useful alternatives to the user, i.e. novel recommendations.

The multiple p-median facility location problem can be applied to recommender systems in order to offer multiple case recommendations to the user, exploiting the parameter (mc) which refers to the number of times that a demand point should be served. The recommendation set presents higher or lower diversity and similarity ratios based on the values of the parameter mc.

Contemporary databases include a huge number of records and thus the exact methods which solve facility location problems may fail to provide a result in reasonable time, whereas heuristic methods that reduce the search space may provide results close to optimal within a shorter time limit. Thus, in this paper, the application of the multiple p-median facility location model to recommender systems is realized through such a specific heuristic method proposed in Panteli *et al.* (2018). However, note that any of the methods that can solve effectively the MPMP can also be used instead. Note that the worst case time complexity of the adopted heuristic method is $O(C^2)$, where C is the number of cases, since it is based on an efficient heuristic biclustering technique.

4. Proposed Approach

4.1. *Application of multiple p-median problem (MPMP) to recommender systems*

In this paper we present the application of the multiple p-median problem to recommender systems as described in FIG. 1.

This facility location problem concerns facilities that serve the demand points. Each facility i can be represented by a column of a matrix and each demand point j by a row or vice versa. Given that columns refer to facilities and rows to demand points, the value $d(i,j)$ indicates the distance between the facility i and the demand point j. The objective of the problem is to select p facilities represented by columns in order to serve each demand point at least mc times, each time from a different facility. The facilities have to be selected in such a way that the total distance between demand points and facilities is the minimum possible. Parameter mc takes values equal or smaller than p. For example, in FIG. 1, a possible solution could be the selection of facilities (columns) i_2 and i_3.

	i1	***i2***	***i3***	i4	i5
j1	0	***15***	***2***	3	18
j2	15	***0***	***1***	12	6
j3	3	***7***	***8***	0	7
j4	5	***8***	***0***	19	14
j5	24	***1***	***6***	18	0
Sum	47	***31***	***17***	52	45

	Case 1	*Case 2*	*Case 3*	Case 4	Case 5	**Target Query**
attr1	dis11	dis12	dis13	dis14	dis15	**v1**
attr2	dis21	dis22	dis23	dis24	dis25	**v2**
attr3	dis31	dis32	dis33	dis34	dis35	**v3**
attr4	dis41	dis42	dis43	dis44	dis45	**v4**
attr5	dis51	dis52	dis53	dis54	dis55	**v5**

FIG. 1. *An example showing how a facility location problem can be applied to recommender systems*

The formulation of MPMP requires defining the decision variables first:

$$x_{ij} = \begin{cases} 1, if\ a\ demand\ point\ j\ is\ served\ by\ a\ facility\ located\ at\ site\ i \\ 0, otherwise \end{cases} \quad (4.1.1)$$

$$y_i = \begin{cases} 1, if\ a\ facility\ is\ located\ at\ candidate\ site\ i \\ 0, otherwise \end{cases} \quad (4.1.2)$$

The MPMP (*multiple p-median problem)*, where p represents the available columns and mc the number of times a demand point should be served, can be defined as follows:

$$Minimize\ \sum_{i \in I} \sum_{j \in J} d_{ij} x_{ij} \quad (4.1.3)$$

$$subject\ to \quad \sum_{i \in I} x_{ij} \geq mc, \quad \forall\, j \in J \tag{4.1.4}$$

$$\sum_{i \in I} y_i = p \tag{4.1.5}$$

$$x_{ij} - y_i \leq 0, \quad \forall\, i \in I; j \in J \tag{4.1.6}$$

$$y_i \in \{0, 1\}, \quad \forall\, i \in I \tag{4.1.7}$$

$$x_{ij} \in \{0, 1\}, \quad \forall\, i \in I; j \in J \tag{4.1.8}$$

The objective function (4.1.3) minimizes the total distance between the demand points and the selected facilities. Constraint (4.1.4) ensures that each demand point is served at least mc times. Note that partial coverage of a demand point by a facility is not considered. Constraint (4.1.5) indicates that p facilities should be located in order to satisfy the demand. Constraint (4.1.6) ensures that no demand point is assigned to a location unless there is an open facility at that location. Constraints (4.1.7) and (4.1.8) refer to the nature of the decision variables.

Recommender systems contain a huge number of cases with specific attributes. Each $Case_i$ can be represented with a column and each attribute $attr_j$ with a row of a matrix. Given a *target query* that a user inserts to the recommender system, all the cases of the database indicate a specific level of dissimilarity with the target query. More specifically, for each attribute $attr_j$ of the $Case_i$ under observation, the cell value dis_{ij} indicates a specific dissimilarity level of the $Case_i$ with the target query for the attribute $attr_j$, given its value v_j. As a result, a matrix can be created containing all the dissimilarity ratios between the target query and all the remaining cases of the database. Also, for each $Case_i$, the total dissimilarity ratio can be measured through the aggregation of the dissimilarity levels for each individual attribute for this case.

It is obvious from FIG. 1 that the MPMP model can be easily applied to recommender systems. The definition of decision variables is formed as follows:

$$x_{ij} = \begin{cases} 1, if\ the\ attr_j\ of the Case_i\ has\ similar\ value\ as\ that\ of\ the\ target\ query \\ 0, otherwise \end{cases} \tag{4.1.9}$$

$$y_i = \begin{cases} 1, if\ the\ Case_i\ is\ selected\ to\ be\ included\ in\ the\ recommendation\ set \\ 0, otherwise \end{cases} \tag{4.1.10}$$

The objective function of the minimization of the total distance between facilities and demand points can be adapted to a recommender system, where the aim is to minimize the *dissimilarity level* between the cases of the recommendation set and the target query, since the aim of the system is to propose cases similar to the target query.

In fact, the selected solutions (biclusters) must contain in total p columns and each row must be served at least by mc columns. The mc parameter can impose a higher or lower similarity to diversity ratio according to the different values it can take. Higher values of the parameter mc lead to higher similarity to diversity ratios of the recommended cases when compared to the target query.

When the number of recommendations is small (p) and thus the parameter mc is small $(mc \leq p)$, the existence of similar cases to the query is more probable. Thus, when mc is small, increasing its value (i.e. from 3 to 5), forces the selected cases to serve each row (attribute) more times and hence forces the cases to be more similar to each other. Therefore, the increase in similarity is high and thus the similarity to diversity ratio also increases.

In contrast, when the number of recommendations (p) is large, the parameter mc can be also large and the existence of similar cases to the query is less probable. As a result, the diversity is high. As the value of mc increases (i.e. from 3 to 10), the selected cases are forced to serve each row (attribute) more times and the diversity is affected. More specifically, the diversity decreases and consequently the similarity to diversity ratio increases.

As a result, the application of the MPMP model to recommender systems offers flexibility and easily controlled transition between diversity and similarity levels, through setting the mc parameter. The latter is one of the major advantages of the proposed approach. Another advantage is the low computation time of the adopted heuristic method which is based on an effective biclustering technique.

In fact, the MPMP has been solved in Panteli *et al.* (2018) with a method that first enumerates candidate biclusters with the use of a biclustering technique presented in Boutsinas (2013b) which accepts as input binary matrices. These biclusters contain mc columns. The total amount of extracted biclusters depends on the minimum support value that is used. In fact, the minimum support measure is defined as the minimum accepted percentage of the number of rows in the whole dataset and determines the minimum number of rows of a bicluster. The greater the minimum support, the higher the number of rows of the extracted biclusters. The method then applies a selection process until p columns are selected. It should be mentioned that for the biclustering phase of the method, any biclustering technique that can be applied to a Boolean matrix can be used instead of the technique presented in Boutsinas (2013b). The method accepts as inputs three parameters, $mc, p, Threshold\ distance$, representing the number of facilities that must serve each demand point, the total number of facilities that should be selected and the threshold distance that is used in order to create a binary matrix $BIN(R, C)$ respectively. The value of $Threshold\ distance$ is set close to the *weighted average* distance. The objective function which represents the total cost (or total distance) selects for each demand point a subset of the p selected facilities including the mc facilities with the lowest cost.

The experimental results, presented in Section 5, demonstrate the case base transformation and the experimental tests.

4.2. *Time complexity of the proposed technique*

Exact methods that solve facility location problems may fail to provide a query result in reasonable time. It was shown theoretically that exponential cases exist within the class of network problems (Megiddo, 1986) during solving linear programming models (e.g. variants of Simplex). On the other hand, most interesting proposed biclustering problems are proved to be NP-complete (Ben-Dor *et al.*, 2003; Cheng & Church, 2000; Tanay *et al.*, 2002).

Thus, the proposed methodology is based on a heuristic method that reduces the search space providing results close to optimal ones within a shorter time limit. The adopted heuristic method for solving the multiple p-median facility location model exhibits an average gap from optimal 2.54% while the average speedup *(speedup is calculated by dividing the CPLEX solution time by the solution time of the proposed method)* is 106, compared to CPLEX for the benchmark instances of OR library. It must be noted that for many instances, the gap and the speed up took values up to 0.72%, and 1865.892 respectively (Panteli *et al.*, 2018).

Also, for large-scale problems up to 1200 nodes, the gap from the optimal solution was up to 0.84% while speed up took values up to 8824. For even larger problems (up to 2000 nodes), given a time limit of 3600 seconds, the proposed method outperformed the best integer solution provided by CPLEX by up to 49%. In fact, without the time limit, the CPLEX optimizer needed more than 4 hours to provide the optimal results whereas the proposed method only took around 2 seconds resulting in a speedup of up to 60000 for these problems. For the problems with more than 2000 and up to 8000 nodes, CPLEX could not even load the data in order to run the model, whereas the proposed method needed less than 18 seconds to produce a result.

The time complexity of the adopted heuristic method is dominated by the time complexity of the used heuristic biclustering algorithm (Boutsinas, 2013b).

It is $O(|R| * |C| * |C_1 \cup C_2 \cup \ldots \cup C_k|)$, where R is the number of characteristics used to describe cases, C is the number of cases and $C_1 \ldots C_k$ are the examined sets of candidate frequent itemsets. Thus, based on the problem formulation, the worst case time complexity of the adopted heuristic method is $O(C^2)$, which is low compared to similar approaches. For instance, the method presented in Hurley & Zhang (2011) is $O(C^3)$ in the worst case, which of course can be reduced. The latter is also true for the proposed method by using more efficient techniques for association rule mining as those in Boutsinas *et al.* (2008) and Han *et al.* (2004).

The running time of the proposed method is in the range of 0.01 to 850 seconds depending on the number of frequent itemsets that are extracted and evaluated from the Apriori algorithm and the number of recommended cases to the user. Our approach was implemented in Java Environment and the experiments were carried out on a PC Intel Core TM i7-4700 CPU (2,40GHz) with 8GB RAM.

5. Experimental Results

5.1. *Case base*

In order to evaluate the proposed methodology, we used the standard benchmark case library for recommender systems, the Travel Case Base[‡] which contains categorical and numerical data. This case base has been appropriately transformed in order to be used by the adopted method for solving the MPMP model, as analyzed below.

5.2. *Data transformation*

Each case of the Travel Case Base, which refers to a specific travel proposal, is represented by a column of the Data Matrix whereas each attribute (such as type, price and number of persons) is represented by a row of the Data Matrix. Each case is characterized by a dissimilarity ratio - when compared to a query as a whole - and a dissimilarity score for each of the attributes.

To facilitate matching on the numeric attributes (price, duration), it is a common approach either to discretize values in some ranges or to scale the difference between query and candidate case to the range of [0,1] (Hurley & Zhang, 2011). We adopted the former approach and we performed experimental tests in order to decide which is the appropriate number of intervals that should be used. However, the "ideal" number of intervals depends on the specific problem under observation and the scope of the research. Therefore, it is common to use functions such as these proposed by Sturges (1926) and Scott (1979) as starting points and then evaluate and adapt the result according to the specific problem and data. As a result, these measures were used and results indicated that it was appropriate to unify the number of intervals suggested by the Sturges measure (Sturges, 1926) into 3 intervals of equal length.

In categorical attributes, the comparison is exact between values of the case under observation and the target query. An important point to note is that we measure similarity, and hence diversity, with respect to the attributes of the target query only. For example, in our case we insert only 7 out of 9 attributes since in most real cases, the target queries inserted by users are often incomplete in the sense that preferred values are specified for only some of the case attributes, thus reducing the number of attributes available for retrieval. This standard methodology has been used by other researchers, e.g. McSherry (2002) and Hurley & Zhang (2011).

The objective is to find cases that present high similarity ratios to the query, while at the same time they are considerably diverse between each other.

In order to apply the proposed biclustering technique for solving the MPMP, the input data is transformed as described below:

1. For *numeric* attributes, each value of the observed case is assigned to a specific range. Case attributes that belong to a specific range take the frequency value of this range.
2. Frequencies of values of *categorical* attributes of the case base are counted and each attribute value of the observed case takes the corresponding frequency value.

[‡] https://ai-cbr.cs.auckland.ac.nz/cases.html

3. For each attribute of the case under observation, if the attribute value is the same as that of the target query, then dissimilarity is set to zero. On the other hand, if it is different, Equation (5.2.1) is used that takes into account the frequencies of the attribute values with respect to the whole data set. More specifically, if A, B are two cases described by m attributes, then the dissimilarity measure is defined as in Huang (1998):

$$d_{x^2}(A,B) = \sum_{j=1}^{m} \left(\left(n_{a_j} + n_{b_j} \right) / \left(n_{a_j} \times n_{b_j} \right) \right) \times \delta\left(a_j, b_j\right) \qquad (5.2.1)$$

where n_{a_j}, n_{b_j} are the numbers of input objects that have values a_j and b_j for attribute j and $\delta\left(a_j, b_j\right) = \begin{cases} 0 \ if \ a_j = b_j \\ 1, if \ a_j \neq b_j \end{cases}$

This measure is commonly used in clustering data with categorical nature (Boutsinas & Papastergiou, 2008). In the Travel Case Base, most of its attributes are categorical and the numeric ones are treated as categorical after their transformation into 3 intervals.

4. For each case of the Case Base, the aggregate dissimilarity of all attributes is counted.
5. The cases of the Case Base are sorted in ascending dissimilarity order and the first w of them are selected (the w most similar ones). The value of w depends on the problem and the characteristics of the Case Base. In some cases (Smyth & McClave, 2001), if p is the retrieval set, the candidate ones are $(2 * p)$. However, the number of candidate cases should be adequate to ensure that the final recommendation set is diverse. In our work, 20 cases were selected for further evaluation.
6. The matrix of w cases should contain integer values in order for the method to be applied, thus it may be necessary to multiply all values by 10 or multipliers of it. However, the method can be easily transformed in order to handle decimal values as well.
7. The proposed method extracts all possible biclusters (potential solutions) according to the threshold value and the minimum support and selects s biclusters that contain in total p columns that represent the selected cases. The minimum support parameter took values from 50% to 70% in our experiments.
8. Initial values of the case base for each selected case are used and the comparison in terms of similarity and diversity is based on exact similarity of nominal attributes and range similarity in numerical attributes.
9. Total similarity and diversity results are measured through the corresponding Equations (5.3.1) & (5.3.2).

5.3. *Experimental tests*

We performed sets of experimental tests for different values of the parameter mc ($mc = 3, 4, 5, 7, 10, 15$) in order to show the effectiveness and flexibility of the proposed methodology. As we have already stated, this parameter refers to the alternative choices which are recommended to the user by a recommender system.

TABLE 1 presents the experimental results of the application of the multiple p-median model to recommender systems. The diversity ratio is calculated as in Smyth & McClave (2001) by the following equation and is defined as the average dissimilarity between all pairs of cases in the case-set:

$$Diversity(c_1, \dots, c_n) = \frac{\sum_{i=1..p} \sum_{j=1..p} \left(1 - Similarity(c_i, c_j)\right)}{\frac{p}{2}(p-1)} \qquad (5.3.1)$$

where p is the number of cases retrieved by the case base in order to find their diversity and similarity respectively.
Similarity is obtained through the following equation:

$$Similarity(t, c) = \frac{\sum_{i=1..p} w_i * sim(t_i, c_i)}{\sum_{i=1..p} w_i} \qquad (5.3.2)$$

where t is the target query and c is the case with which the comparison is done.

The Travel Case Base contains 1024 cases but no real user queries. Thus, we used for our experiments the leave-one-out approach in which each case is removed from the Case Base in turn, and the values of its attributes are considered as a query to the proposed approach. We removed from each query, as we already stated, some of its features following the common practice in the literature (Hurley & Zhang, 2011; McSherry, 2002).

For each target query, we performed experimental tests for different values of the p parameter. In fact, experimental tests were performed for $p = 5, 10, 15$ and 20. A total mc out of the cases (columns) should match the attribute values (rows) of the target query. The extracted biclusters (potential solutions) contain mc columns (cases). As a result, from the definition of biclusters, cases (columns) that belong to the same bicluster present similar behavior across a subset of attributes (rows).

As the value of mc increases, we expect higher similarity to diversity ratios of the recommended cases when compared to the target query. When the recommendation set is limited and consequently the parameter mc is small since $mc \leq p$, the existence of similar cases to the query is more probable. Thus, when mc is small, as it increases (i.e. from 3 to 4 or 5), it forces the selected cases to serve each row (attribute) more times and hence forces the cases to be more similar to each other. Therefore, the increase in similarity is high and thus the similarity to diversity ratio increases. This is represented in TABLE 1 and in FIG. 2 that illustrates the results of similarity and diversity measures and the ratios of similarity to diversity when the recommendation set is small ($p = 5$). In fact, the average similarity to diversity ratio of recommended cases when $mc = 5$ is higher than the average ratio when $mc = 4$ and the corresponding ratio when $mc = 4$ is greater than the one when $mc = 3$. The average similarity is high enough as we expected. When the value of mc increases from 3 to 5, the average similarity also increases (from 0.73169 to 0.73295 and 0.73321 correspondingly) as shown in TABLE 1. This high value of average similarity, due to the trade-off between similarity and diversity, implies a lower value of diversity.

In contrast, when the size of the recommendation list is large ($p = 10, 15, 20$), the existence of similar cases to the query is less probable and thus the diversity is high. In general, only a limited number of cases present a high similarity ratio when compared to a specific query. The number depends on the nature of the data contained in a specific Case Base. Thus, the number of cases that are relevant to the query is limited. Indeed, from the experimental tests that we performed and the evaluation measures that we used (see Section 5.4.2), the Travel Case Base contains only a limited number of cases that match each query. As a result, when the recommendation list is large, increasing mc (e.g. from 3 to 10), tends to force the selected cases to serve each row (attribute) more times which affects the diversity measure. More specifically, the diversity decreases and consequently the similarity to diversity ratio increases. This is shown in TABLE 1 for the corresponding values of $p = 10{,}15{,}20$ and $mc = 3{,}5{,}7{,}10{,}15{,}20$ and in FIG. 3, FIG. 4 and FIG. 5.

TABLE 1 *Experimental Results*

mc	**p**	**Diversity**	**Similarity**	**Similarity to Diversity Ratio**
3	5	**0.33783**	**0.73169**	2.166
4	5	0.31407	0.73295	2.334
5	5	0.25923	0.73321	2.828
3	10	0.36108	0.70065	1.940
5	10	0.35329	0.70068	1.983
7	10	0.33663	0.69950	2.078
10	10	0.30170	0.69990	2.320
5	15	0.37412	0.67896	1.815
10	15	0.35160	0.66820	1.900
15	15	0.34350	0.66370	1.932
5	20	0.42599	0.63301	1.486
10	20	0.40553	0.63494	1.566
20	20	0.38848	0.62245	1.602

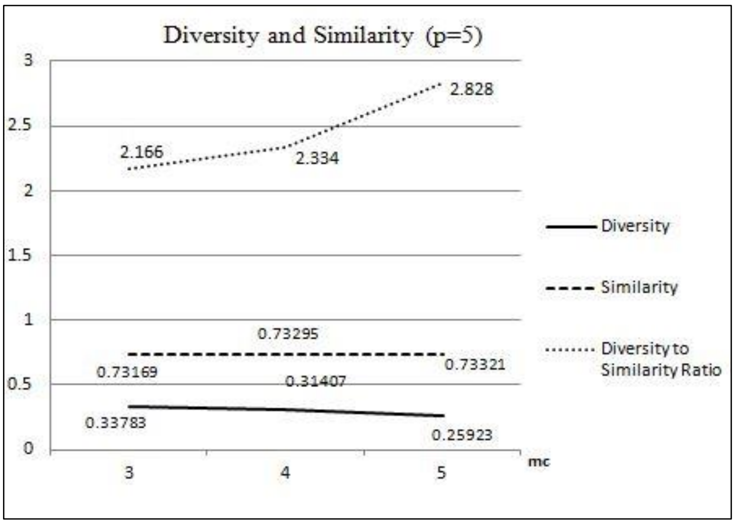

FIG. 2. *Graph representation for Similarity and Diversity results (p=5)*

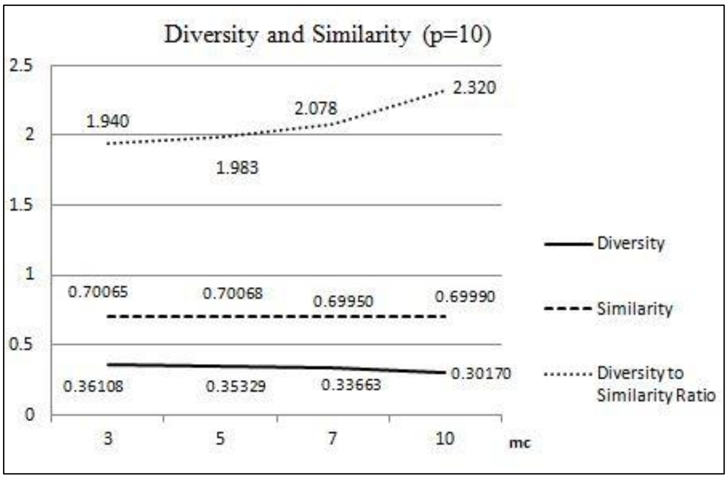

FIG. 3. *Graph representation for Similarity and Diversity results (p=10)*

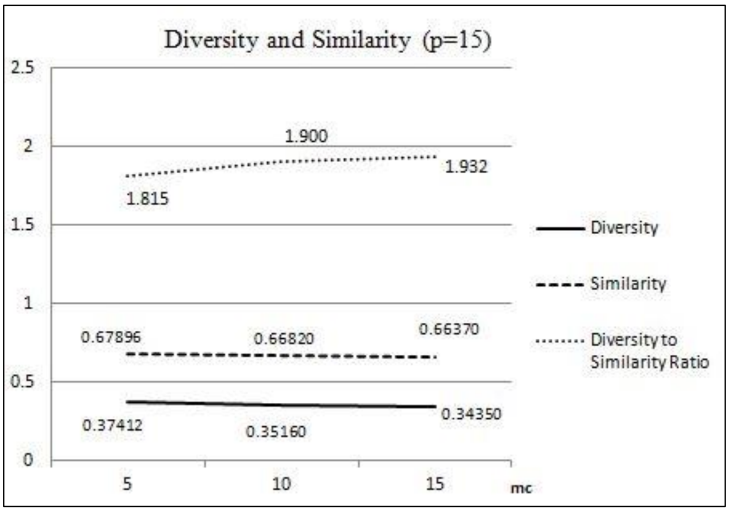

FIG. 4. *Graph representation for Similarity and Diversity results (p=15)*

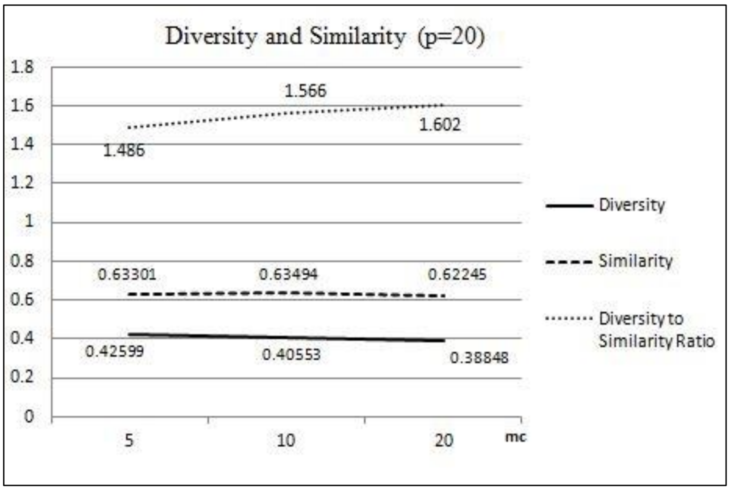

FIG. 5. *Graph representation for Similarity and Diversity results (p=20)*

5.4. *Comparison with other methods*

As mentioned earlier, the leave-one-out method is adopted for the experimental tests. However, in order to perform a comparison analysis, we also performed tests by randomly selecting a subset of cases as queries, as in Smyth & McClave (2001). More specifically, 400 cases were randomly selected out of the whole set of 1024 cases in order to form queries for $p = 5$ and the remaining cases forming the Case Base. The results for both approaches are presented in TABLE 2.

The average diversity (TABLE 2) of the selected cases, calculated by Equation (5.3.1) with $mc = 3$, is 0.33783 and thus it is quite satisfactory when compared to the relative results of diversity presented in Smyth & McClave (2001). As shown in TABLE 2, our implementation, outperforms the Standard and Random technique in terms of diversity, whose values range from 0.289 to 0.326 (Smyth & McClave, 2001) respectively and performs well against Bounded Greedy (0.375) and Greedy technique (0.458) for $p = 5$, where p is the retrieval set size. However, when $mc = 4$ and $mc = 5$ and the similarity is higher, due to the trade-off between the two measures, the diversity ratio decreases to 0.31407 and 0.25923 respectively.

Moreover, the proposed method reports even better results in terms of diversity when tests are performed by randomly selecting 400 cases as queries, as in Smyth & McClave (2001). As shown in TABLE 2, the similarity value in this case is 0.69 and the diversity value is 0.41.

Here, it should be mentioned that the comparison is performed for $p = 5$, which refers to the number of recommended cases to the user. This is due to the fact that the exact values of diversity and similarity for the alternative diversity preserving techniques and for different values of p are not available in the literature (Smyth & McClave, 2001). However, in Smyth & McClave (2001), it has been reported that the value of 0.4 for the diversity measure is achieved for the Bounded Greedy method when $p = 10$, for the Random method when $p = 23$ and for the Standard method when $p = 46$, whereas for the proposed method it is achieved when p ranges between 15 and 20, as shown in TABLE 1.

TABLE 2 *Effectiveness of the proposed method with respect to alternative techniques in literature*

		Standard	**Bounded Greedy**	**Random**	**Greedy**	**Proposed Method (Leave-One-Out)**	**Proposed Method (400 Queries)**
		S	BG	R	G	PM	PM
p=5	**Similarity**	0.78	0.753	0.748	0.7	0.73169	0.69
	Diversity	0.289	0.375	0.326	0.458	0.33783	0.41

5.4.1. *Evaluation of the proposed method*

Since there is a trade-off between similarity and diversity, which means that the increase of diversity is usually achieved at the expense of similarity, it would be useful to propose an approach that fully preserves the degree of similarity while at the same time achieves higher levels of diversity. Relative benefit is a metric proposed in the literature ((Smyth & McClave, 2001)) in order to evaluate similar methods regarding the trade-off between the two measures. The relative benefit, in general, is calculated by measuring increases in diversity relative to decreases in similarity with respect to the Standard method. However, it seems that this metric cannot be adopted to compare any set of such methods while it does not measure the trade-off directly.

For the latter, note that the adoption of the Standard method as the base of comparison is not proved to be the best choice. Alternatively, one could consider the Greedy method as the base of comparison by measuring decreases in diversity relative to increases in similarity with respect to the Greedy method. In this case, the evaluation results for the various diversity preserving techniques compared in Smyth & McClave (2001) are different.

For the former, note that if there exists a method that outperforms the Standard method in terms of both diversity and similarity, the value of the relative benefit for this method is negative. A method that outperforms the Standard Method in terms of both diversity and similarity should be ranked in the highest position than the others. However, since the greater the value of relative benefit the better the ranking of an evaluated method, a method evaluated with a negative value of relative benefit will be ranked lower, which is an incorrect evaluation.

In this paper, we introduce the Gap metric that can be adopted to compare any set of such methods since it measures the trade-off directly. It aims to measure the gap between similarity and diversity with respect to similarity. It is calculated by dividing the difference between the value of similarity and diversity by the similarity. Thus, a method is good if the Gap metric is low, i.e. if the difference between similarity and diversity is low while similarity is high.

TABLE 3 *Relative Benefit of the proposed method in comparison to alternatives*

		Standard	**Bounded Greedy**	**Rando m**	**Greedy**	**Proposed Method (Leave-One-Out)**	**Proposed Method (400 Queries)**
p=5	**Similarity**	0.78	0.753	0.748	0.7	0.73169	0.69
	Diversity	0.289	0.375	0.326	0.458	0.33783	0.41
Gap Metric		**0.63**	**0.502**	**0.564**	**0.346**	**0.538**	**0.406**

Based on the Gap metric, we evaluated the alternative methods. The corresponding results of the metric that are presented in TABLE 3 show that the proposed algorithm (when the leave-one-out method is used) outperforms both the Standard and the Random method and is quite close to the Bounded Greedy technique. However, when the method of the randomly selected 400 queries is adopted, as in Smyth & McClave (2001), the evaluation metric results show that the proposed method outperforms all the alternative methods apart from the Greedy method which however, exhibits a very high time complexity (Smyth & McClave, 2001). Thus, in general, the proposed algorithm performs well when it is compared to the alternative diversity preserving techniques, while at the same time it has a lower time complexity.

5.4.2. *Precision analysis*

In order to further evaluate the proposed method, we performed a precision analysis based on the method used in Hurley & Zhang (2011). Precision is widely used in information retrieval in order to evaluate a system's accuracy. It consists of retaining a set T_u of "ground-truth" cases that are known to be relevant to the user inserted query (Hurley & Zhang, 2011) and evaluating the system's ability to retrieve these items.

However, in contrast to collaborative filtering techniques, in case-based recommendation we do not have information on which set of cases best matches a particular query (Hurley & Zhang, 2011). Thus, we need to find a set of cases that are relevant to each query. In order to achieve this, we used a query-case similarity function, as proposed in Hurley & Zhang (2011):

$$\epsilon sim_{(q,c)} \tag{5.4.2.1}$$

Where ϵ is a small scaling parameter $\epsilon \in [0,1]$, since only a limited number of cases match each given query. The $sim_{(q,c)}$ is a similarity function of the case to the given query. We used this function in the same way we used it in Section 5.2, for numeric and categorical attributes. Thus, by applying Equation (5.2.1) a relevant set T_u was obtained for each case of the case-base. The proposed recommendation strategy was then tested for $p = 5$ against the relevant cases for each case in turn, on a set of over 500 randomly generated queries.

The adopted metric for precision evaluation is defined in (Sarwar *et al.*, 2000b) and was also used in Hurley & Zhang (2011):

$$Precision = \frac{|T_u \cap R_u|}{p} \tag{5.4.2.2}$$

where T_u is a set of items known to be relevant to the user, R_u is the recommended set of items for user u by the proposed retrieval strategy and p is the number of recommended cases to the user by the retrieval strategy.

The precision was normalized by dividing it by the scaling parameter ϵ. The experimental tests indicated that the precision of the proposed method is almost 80%, a value approaching the precision of the algorithm that constructs the recommendation set only based on the similarity of cases to the given query and better than the precision obtained by the methods tested in Hurley & Zhang (2011) which varied from 60-75%.

We have also conducted experimental tests regarding the precision for greater values of p. However, users are more likely to be overwhelmed and confused by recommendation lists that contain a large number of items, as stated in Pu *et al.* (2011). Also, the limited size of the recommendation list is essential for small display devices such as mobile phones. Moreover, precision deteriorates by the increase in the number of recommended items, p, because of less matches (Gunawardana & Shani, 2009; Peker & Kocyigit, 2016; Sarwar *et al.*, 2000a).

Indeed, the precision measure indicated that when the value of p increases, the percentage cases that are similar to the query decreases. Therefore, it is not really useful to calculate the precision measure for larger values of p.

6. Conclusions

The application of a facility location model to recommender systems, first studied in this paper, provides effectiveness and flexibility in terms of similarity of the recommending set when compared to the target query and diversity between the recommended cases with each other.

The mc parameter constitutes a regulator for the trade-off between similarity and diversity of the recommendation set. When the size of the recommendation set is small, the existence of similar cases to the target query is more probable, thus the increase of the mc parameter forces the selection of cases that serve each row (attribute) more times and hence these cases are more similar to each other, resulting in a similarity increase and consequently in the increase of similarity to diversity ratio. When the size of the recommendation set is large, the existence of similar cases to the target query is less probable and thus diversity is high. However, as the mc becomes higher, it tends to force the selected cases to serve each row (attribute) more times and this leads to diversity decrease and to increase of the similarity to diversity ratio. Moreover, the proposed approach performs well when compared to the alternative diversity preserving techniques, while at the same time it exhibits very fast computation times with respect to these approaches.

Acknowledgement

The authors wish to thank Aristotelis Kompothrekas, PhD candidate in Patras University, for his valuable contribution in performing the experimental tests.

REFERENCES

ABBASSI, Z., MIRROKNI, V. S., & THAKUR, M. (2013). *Diversity maximization under matroid constraints.* In: Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining.

ADOMAVICIUS, G., & KWON, Y. (2012). Improving aggregate recommendation diversity using ranking-based techniques. *IEEE transactions on knowledge and data engineering, 24*(5), 896-911.

AGRAWAL, R., GOLLAPUDI, S., HALVERSON, A., & IEONG, S. (2009). *Diversifying search results.* In: Proceedings of the second ACM international conference on web search and data mining.

ALP, O., ERKUT, E., & DREZNER, Z. (2003). An efficient genetic algorithm for the p-median problem. *Annals of Operations Research*, *122*(1), 21-42. doi: 10.1023/A:1026130003508

AVELLA, P., BOCCIA, M., SALERNO, S., & VASILYEV, I. (2012). An aggregation heuristic for large scale p-median problem. *Computers & Operations Research, 39*(7), 1625-1632. doi: 10.1016/j.cor.2011.09.016

AYTEKIN, T., & KARAKAYA, M. Ö. (2014). Clustering-based diversity improvement in top-N recommendation. *Journal of Intelligent Information Systems, 42*(1), 1-18. doi: 10.1007/s10844-013-0252-9

BEASLEY, J. E. (1985). A note on solving large p-median problems. *European Journal of Operational Research, 21*(2), 270-273. doi: 10.1016/0377-2217(85)90040-2

BEN-DOR, A., CHOR, B., KARP, R., & YAKHINI, Z. (2003). Discovering local structure in gene expression data: the order-preserving submatrix problem. *Journal of computational biology, 10*(3-4), 373-384. doi: 10.1089/10665270360688075

BOBADILLA, J., SERRADILLA, F., & HERNANDO, A. (2009). Collaborative filtering adapted to recommender systems of e-learning. *Knowledge-Based Systems*, *22*(4), 261-265.

BOIM, R., MILO, T., & NOVGORODOV, S. (2011). *Diversification and refinement in collaborative filtering recommender.* In: Proceedings of the 20th ACM international conference on Information and knowledge management.

BOUSBAHI, F., & CHORFI, H. (2015). MOOC-Rec: a case based recommender system for MOOCs. *Procedia-Social and Behavioral Sciences, 195*, 1813-1822. doi: 10.1016/j.sbspro.2015.06.395

BOUTSINAS, B. (2013a). Machine-part cell formation using biclustering. *European Journal of Operational Research, 230*(3), 563-572. doi: 10.1016/j.ejor.2013.05.007

BOUTSINAS, B. (2013b). A new biclustering algorithm based on association rule mining. *International Journal on Artificial Intelligence Tools, 22*(03), 1350017. doi: 10.1016/j.ejor.2013.05.007

BOUTSINAS, B., & PAPASTERGIOU, T. (2008). On clustering tree structured data with categorical nature. *Pattern recognition, 41*(12), 3613-3623.

BOUTSINAS, B., SIOTOS, C., & GEROLIMATOS, A. (2008). Distributed mining of association rules based on reducing the support threshold. *International Journal on Artificial Intelligence Tools, 17*(06), 1109-1129. doi: 10.1142/S0218213008004321

BOZKAYA, B., ZHANG, J., & ERKUT, E. (2002). An efficient genetic algorithm for the p-median problem. *Facility location: Applications and theory*, 179-205.

BRADLEY, K., & SMYTH, B. (2001). *Improving recommendation diversity.* In: Proceedings of the Twelfth Irish Conference on Artificial Intelligence and Cognitive Science, Maynooth, Ireland.

BUSYGIN, S., PROKOPYEV, O., & PARDALOS, P. M. (2008). Biclustering in data mining. *Computers & Operations Research, 35*(9), 2964-2987. doi: 10.1016/j.cor.2007.01.005

CHEN, H.-C., & CHEN, A. L. (2005). A music recommendation system based on music and user grouping. *Journal of Intelligent Information Systems, 24*(2), 113-132. doi: 10.1007/s10844-005-0319-3

CHENG, Y., & CHURCH, G. M. (2000). *Biclustering of expression data.* In: Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology.

CHOI, S.-M., & HAN, Y.-S. (2010). *A content recommendation system based on category correlations.* In: Computing in the Global Information Technology (ICCGI), 2010 Fifth International Multi-Conference on.

CLARKE, C. L., *et al.* (2008). *Novelty and diversity in information retrieval evaluation.* In: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval.

DAS SARMA, A., GOLLAPUDI, S., & IEONG, S. (2008). *Bypass rates: reducing query abandonment using negative inferences.* In: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining.

DENSHAM, P. J., & RUSHTON, G. (1992). A more efficient heuristic for solving large p-median problems. *Papers in regional science, 71*(3), 307-329. doi: 10.1111/j.1435-5597.1992.tb01849.x

DEODHAR, M., & GHOSH, J. (2007). *A framework for simultaneous co-clustering and learning from complex data.* In: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining.

DHILLON, I. S. (2001). *Co-clustering documents and words using bipartite spectral graph partitioning.* In: Proceedings of the 7th ACM SIGKDD international conference on Knowledge discovery and data mining.

DREZNER, Z., & HAMACHER, H. W. (2001). *Facility location: applications and theory*: Springer Science & Business Media.

FESENMAIER, D. R., *et al.* (2003). *DIETORECS: Travel advisory for multiple decision styles*.

FITZSIMMONS, J. A., & ALLEN, L. A. (1983). A warehouse location model helps Texas comptroller select out-of-state audit offices. *Interfaces, 13*(5), 40-46. doi: 10.1287/inte.13.5.40

FLEDER, D., & HOSANAGAR, K. (2009). Blockbuster culture's next rise or fall: The impact of recommender systems on sales diversity. *Management Science, 55*(5), 697-712.

FLEDER, D. M., & HOSANAGAR, K. (2007). *Recommender systems and their impact on sales diversity.* In: Proceedings of the 8th ACM conference on Electronic commerce.

GARY, M. R., & JOHNSON, D. S. (1979). Computers and Intractability: A Guide to the Theory of NP-completeness: WH Freeman and Company, New York.

GOLBECK, J. (2006). *Generating predictive movie recommendations from trust in social networks*. In: International Conference on Trust Management.

GUNAWARDANA, A., & SHANI, G. (2009). A survey of accuracy evaluation metrics of recommendation tasks. *Journal of Machine Learning Research, 10*(Dec), 2935-2962.

HAKIMI, S. L. (1964). Optimum locations of switching centers and the absolute centers and medians of a graph. *Operations Research, 12*(3), 450-459. doi: 10.1287/opre.12.3.450

HAN, J., PEI, J., YIN, Y., & MAO, R. (2004). Mining frequent patterns without candidate generation: A frequent-pattern tree approach. *Data mining and knowledge discovery, 8*(1), 53-87. doi: 10.1023/B:DAMI.0000005258.31418.83

HANJOUL, P., & PEETERS, D. (1985). A comparison of two dual-based procedures for solving the p-median problem. *European Journal of Operational Research, 20*(3), 387-396. doi: 10.1016/0377-2217(85)90012-8

HANSEN, P., & MLADENOVIĆ, N. (1997). Variable neighborhood search for the p-median. *Location Science, 5*(4), 207-226. doi: 10.1016/S0966-8349(98)00030-8

HARTIGAN, J. A. (1972). Direct clustering of a data matrix. *Journal of the american statistical association, 67*(337), 123-129. doi: 10.1080/01621459.1972.10481214

HERLOCKER, J. L., KONSTAN, J. A., TERVEEN, L. G., & RIEDL, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst., 22*(1), 5-53. doi: 10.1145/963770.963772

HUANG, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data mining and knowledge discovery, 2*(3), 283-304. doi: 10.1023/A:1009769707641

HURLEY, N., & ZHANG, M. (2011). Novelty and diversity in top-n recommendation--analysis and evaluation. *ACM Transactions on Internet Technology (TOIT), 10*(4), 14. doi: 10.1145/1944339.1944341

KARIV, O., & HAKIMI, S. L. (1979). An algorithmic approach to network location problems. II: The p-medians. *SIAM Journal on Applied Mathematics, 37*(3), 539-560. doi: 10.1137/0137041

LAZZERONI, L., & OWEN, A. (2002). Plaid models for gene expression data. *Statistica sinica*, 61-86.

LEE, H. J., & PARK, S. J. (2007). MONERS: A news recommender for the mobile web. *Expert Systems with Applications, 32*(1), 143-150.

LEE, J. S., & LEE, J. C. (2007). *Context awareness by case-based reasoning in a music recommendation system.* In: International Symposium on Ubiquitious Computing Systems.

LIU, J., & WANG, W. (2003). *OP-Cluster: Clustering by Tendency in High Dimensional Space*. In: Proceedings of the Third IEEE International Conference on Data Mining.

LORENZI, F., RICCI, F., TOSTES, R., & BRASIL, R. (2005). Case-based recommender systems: A unifying view. *Lecture notes in computer science, 3169*, 89.

MADEIRA, S. C., & OLIVEIRA, A. L. (2004). Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB), 1*(1), 24-45. doi: 10.1109/TCBB.2004.2

MARANZANA, F. (1964). On the location of supply points to minimize transport costs. *OR*, 261-270. doi: 10.2307/3007214

MCSHERRY, D. (2002). Diversity-Conscious Retrieval. In S. Craw & A. Preece (Eds.), *Advances in Case-Based Reasoning: 6th European Conference, ECCBR 2002 Aberdeen, Scotland, UK, September 4–7, 2002 Proceedings* (pp. 219-233). Berlin, Heidelberg: Springer Berlin Heidelberg.

MEGIDDO, N. (1986). *On the complexity of linear programming*: IBM Thomas J. Watson Research Division.

MITROPOULOS, P., MITROPOULOS, I., & GIANNIKOS, I. (2013). Combining DEA with location analysis for the effective consolidation of services in the health sector. *Computers & Operations Research, 40*(9), 2241-2250.

MITROPOULOS, P., MITROPOULOS, I., GIANNIKOS, I., & SISSOURAS, A. (2006). A biobjective model for the locational planning of hospitals and health centers. *Health Care Management Science, 9*(2), 171-179.

MLADENOVIĆ, N., BRIMBERG, J., HANSEN, P., & MORENO-PÉREZ, J. A. (2007). The p-median problem: A survey of metaheuristic approaches. *European Journal of Operational Research, 179*(3), 927-939. doi: 10.1016/j.ejor.2005.05.034

MUCHERINO, A., & URTUBIA, A. (2010). *Consistent Biclustering and Applications to Agriculture.* In: Industrial Conference on Data Mining-Workshops.

MUSTO, C., & SEMERARO, G. (2015). *Case-based Recommender Systems for Personalized Finance Advisory.* In: FINREC.

PANTELI, A., BOUTSINAS, B., & GIANNIKOS, I. (2018). On solving the multiple p-median problem in large scale data based on biclustering. *Operational Research, to appear*.

PEKER, S., & KOCYIGIT, A. (2016). *An Adjusted Recommendation List Size Approach for Users' Multiple Item Preferences.* In: International Conference on Artificial Intelligence: Methodology, Systems, and Applications.

PERUGINI, S., GONÇALVES, M. A., & FOX, E. A. (2004). Recommender Systems Research: A Connection-Centric Survey. *Journal of Intelligent Information Systems, 23*(2), 107-143. doi: 10.1023/b:jiis.0000039532.05533.99

PRAKASH, P. (2015). Decision support system in heart disease diagnosis by case based recommendation. *Int. J. Sci. Technol. Res, 4*, 51-55.

PREMCHAISWADI, W., POOMPUANG, P., JONGSWAT, N., & PREMCHAISWADI, N. (2013). *Enhancing Diversity-Accuracy Technique on User-Based Top-N Recommendation Algorithms.* In: Computer Software and Applications Conference Workshops (COMPSACW), 2013 IEEE 37th Annual.

PU, P., *et al.* (2011). Usability guidelines for product recommenders based on example critiquing research *Recommender systems handbook* (pp. 511-545): Springer.

REVELLE, C. S., & SWAIN, R. W. (1970). Central facilities location. *Geographical Analysis, 2*(1), 30-42. doi: 10.1111/j.1538-4632.1970.tb00142.x

RICCI, F., ROKACH, L., & SHAPIRA, B. (2011). Introduction to recommender systems handbook *Recommender systems handbook* (pp. 1-35): Springer.

ROSENWEIN, M. B. (1994). Discrete location theory, edited by P. B. Mirchandani and R. L. Francis, John Wiley & Sons, New York, 1990, 555 pp. *Networks, 24*(2), 124-125. doi: 10.1002/net.3230240212

SARWAR, B., KARYPIS, G., KONSTAN, J., & RIEDL, J. (2000a). *Analysis of recommendation algorithms for e-commerce.* In: Proceedings of the 2nd ACM conference on Electronic commerce.

SARWAR, B., KARYPIS, G., KONSTAN, J., & RIEDL, J. (2000b). Application of dimensionality reduction in recommender system-a case study: Minnesota Univ Minneapolis Dept of Computer Science.

SARWAR, B., KARYPIS, G., KONSTAN, J., & RIEDL, J. (2002). *Incremental singular value decomposition algorithms for highly scalable recommender systems.* In: Fifth International Conference on Computer and Information Science.

SCOTT, D. W. (1979). On optimal and data-based histograms. *Biometrika, 66*(3), 605-610.

SHIH, D.-H., YEN, D. C., LIN, H.-C., & SHIH, M.-H. (2011). An implementation and evaluation of recommender systems for traveling abroad. *Expert Systems with Applications, 38*(12), 15344-15355.

SMYTH, B., & MCCLAVE, P. (2001). *Similarity vs. Diversity*. In: Proceedings of the 4th International Conference on Case-Based Reasoning: Case-Based Reasoning Research and Development.

STURGES, H. A. (1926). The choice of a class interval. *Journal of the american statistical association, 21*(153), 65-66.

TANAY, A., SHARAN, R., & SHAMIR, R. (2002). Discovering statistically significant biclusters in gene expression data. *Bioinformatics, 18*(suppl 1), S136-S144. doi: 10.1093/bioinformatics/18.suppl_1.S136

TEITZ, M. B., & BART, P. (1968). Heuristic methods for estimating the generalized vertex median of a weighted graph. *Operations Research, 16*(5), 955-961. doi: 10.1287/opre.16.5.955

TIIHONEN, J., & FELFERNIG, A. (2017). An introduction to personalization and mass customization. *Journal of Intelligent Information Systems, 49*(1), 1-7. doi: 10.1007/s10844-017-0465-4

TRANG TRAN, T. N., ATAS, M., FELFERNIG, A., & STETTINGER, M. (2017). An overview of recommender systems in the healthy food domain. *Journal of Intelligent Information Systems*. doi: 10.1007/s10844-017-0469-0

UNGAR, L., & FOSTER, D. P. (1998). A formal statistical approach to collaborative filtering. *CONALD'98*.

YANG, J., WANG, W., WANG, H., & YU, P. (2002). *δ-clusters: capturing subspace correlation in a large data set.* In: Data Engineering, 2002. Proceedings. 18th International Conference on.

ZHANG, M., & HURLEY, N. (2008). *Avoiding monotony: improving the diversity of recommendation lists.* In: Proceedings of the 2008 ACM conference on Recommender systems.

ZHENG, K., *et al.* (2017). A survey of query result diversification. *Knowledge and Information Systems, 51*(1), 1-36. doi: 10.1007/s10115-016-0990-4

ZHOU, T., *et al.* (2010). Solving the apparent diversity-accuracy dilemma of recommender systems. *Proceedings of the National Academy of Sciences, 107*(10), 4511-4515. doi: 10.1073/pnas.1000488107

ZHOU, T., *et al.* (2009). Accurate and diverse recommendations via eliminating redundant correlations. *New Journal of Physics, 11*(12), 123008. doi: 10.1088/1367-2630/11/12/123008

ZIEGLER, C.-N., MCNEE, S. M., KONSTAN, J. A., & LAUSEN, G. (2005). *Improving recommendation lists through topic diversification.* In: Proceedings of the 14th international conference on World Wide Web.