International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Evaluating Characteristics of k-Most User Influenced Products with Expected Potential Customers Strength

Kadiyam. Ramaiah[1], P. Anitha Rani[2]

[1]Computer Science Engineering, Guntur Engineering College, Guntur, India

Abstract: ***Product planning is an vital phase, calculating factors influencing the outcomes has to established to the core to see the desired potentials, we addressed a problem of production plans, named k-most demanding products (k-MDP) discovering. Given a set of customers demanding a certain type of products with multiple attributes, a set of existing products of the type, a set of candidate products that can be offered by a company, and a positive integer k, we want to help the company to select k products from the candidate products such that the expected number of the total customers for the k products is maximized. We show the problem is NP hard when the number of attributes for a product is 3 or more. One greedy algorithm is proposed to find approximate solution for the problem. We also attempt to find the optimal solution of the problem by estimating the upper bound of the expected number of the total customers for a set of k candidate products for reducing the search space of the optimal solution. An exact algorithm is then provided to find the optimal solution of the problem by using this pruning strategy. The experiment results demonstrate that both the efficiency and memory requirement of the exact algorithm are comparable to those for the greedy algorithm, and the greedy algorithm is well scalable with respect to k.***

Keywords: Algorithms for data and knowledge management, decision support, performance evaluation of algorithm and systems, query processing.

1. Introduction

Microeconomics is a branch of economics, which studies how customers and producers make decisions and how they interact in markets [10]. Customer preference is an important factor in making decisions of product sales, which thus becomes one major concern in microeconomics. Kleinberg et al. [5] pointed out that, when making production plans or marketing strategies, companies usually need to identify one with the highest utility or value. They claimed that the utility or value of a production plan can be modelled as a function that reflects the interaction of the company with other agents such as customers and competitors. Inspired by Kleinberg et al. [5] to take competition into consideration, the problem studied in this paper is to identify the production plan with the highest utility for a company, where the utility of a production plan is evaluated according to the expected number of the total customers for the selected products in the plan.

Consider the scenario of the rental property market at a city as shown in Fig. 1, where the distance to a nearby school and to a subway station are main requirements of the customers demanding a rental property. To make a good marketing decision, a rental company has collected the requirements of the distance to a school and to a subway station from the customers, which are represented by the circular points. The squared points represent the geo-graphic properties of the existing rental properties. Now assume the rental company owns a set of properties whose geographic locations are represented by the triangular points. The manager of the rental company wants to select k properties to compete with the existing rental properties for rental. For getting most profit, an obvious strategy is to get more expected number of the total customers for the k chosen properties. It is assumed that each customer will choose one of the rental properties satisfying his/her requirements. When more than one rental property satisfies the requirements of a customer, the customer will choose one of the properties according to his/her implicit preference. For the sake of simplicity, it is assumed that a customer will choose any qualified rental property with equal probability [20].

Suppose that k is set to 3 and the three properties, denoted cp2, cp3, and cp4in Fig. 1, are selected for rental, the set of available rental properties will become {ep1, ep2, ep3, cp2, cp3, cp4}. Because the customerc1 is satisfied by the existing rental propertiesep1 and the two new rental properties cp2 and cp3, the probability thatc1 will choosecp2 is 1/3. Consequently, the expected number of the customers for cp2 is estimated by

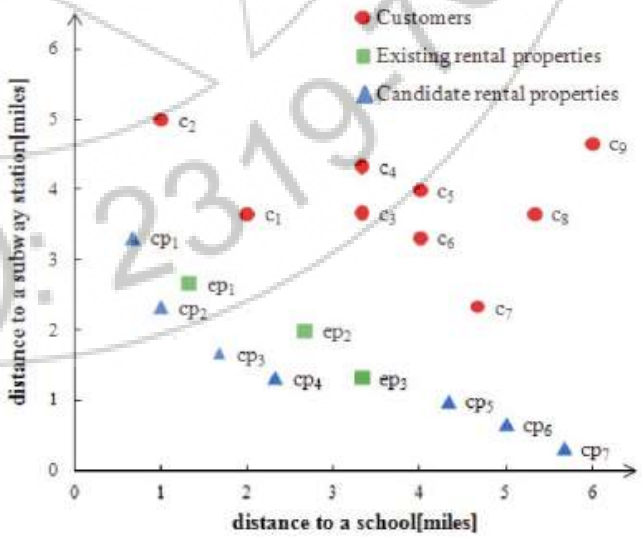

Figure 1: An example for the k-MDP discovering.

adding the probabilities for each customer choosing cp_2 as follows:

$$\left(\frac{1}{3}+\frac{1}{1}+\frac{1}{6}+\frac{1}{6}+\frac{1}{6}+\frac{1}{6}+\frac{1}{5}+\frac{1}{6}+\frac{1}{6}\right) = 2.53$$

Similarly, the expected number of the customers for cp_3 and cp_4, respectively, are estimated as follows:

$$\left(\frac{1}{3}+0+\frac{1}{6}+\frac{1}{6}+\frac{1}{6}+\frac{1}{6}+\frac{1}{5}+\frac{1}{6}+\frac{1}{6}\right) = 1.53 \text{ and}$$

$$\left(\frac{1}{3}+0+\frac{1}{6}+\frac{1}{6}+\frac{1}{6}+\frac{1}{6}+\frac{1}{5}+\frac{1}{6}+\frac{1}{6}\right) = 1.2.$$

Therefore, the expected number of the total customers for {cp2, cp3, cp4} is 5:26(= 2:53 + 1:53 + 1:2), which is the highest among all the possible sets consisting of three candidate rental properties.

According to the above motivating application, we define the problem of k-most demanding products (k-MDP) discovering. Given a set of customers demanding a certain type of products with multiple attributes, a set of existing products of the type, and a set of candidate products that can be offered by a company, we want to help the company to select k products from these candidate products such that the expected number of the total customers for the k products is maximized.

Let EP and CP denote the set of existing products and the set of candidate products, respectively. In addition, kCP denotes the set of k products chosen from CP, cp denotes a candidate product in k CP, and c denotes a customer whose requirements are satisfied by cp. The probability for c choosing cp is inverse proportional to the total number of products, including EP and kCP, which satisfy c. Therefore, the expected number of the customers for cp is influenced not only by the number of customers satisfied by cp but also the total number of other products that satisfy the same set of customers. Notice that it is possible that the products in kCP will compete with each other if they satisfy the same set of customers. Consequently, no simple strategy can be applied to find the set of k candidate products with the largest expected number of the total customers. How to provide an efficient and effective algorithm for solving the k-MDP discovering problem is the goal of this paper.

The contributions of this paper are summarized as follows:

1. We formulate the problem of the k-MDP discovering to be an optimization problem of an objective function
2. We prove the k-MDP discovering problem is NP hard when the number of attributes for a product is3 or more.
3. Two greedy algorithms are proposed to find approximate solutions for the k-MDP discovering problem.
4. We also attempt to find the optimal solution of the problem by estimating the upper and lower bounds of the expected number of the total customers for a set of k candidate products for reducing the search space of the optimal solution. Two exact algorithms are then proposed to find the optimal solution of the problem by using the pruning strategies.
5. A systematic performance study is performed to verify the effectiveness and the efficiency of the proposed algorithms.

The remainder of the paper is organized as follows: The related works are surveyed in Section 2. The form al problem definition and time complexity analysis of the k-MDP discovering problem are given in Section 3. An index structure and two greedy algorithms for solving the problem are described in Section 4. Section 5 introduces two algorithms for finding the optimal solution of the problem with pruning strategies. The performance evaluation on the proposed algorithms is reported in Section 6.Finally, in Section 7, we conclude this paper and show the directions for our future studies.

2. Related Work

Microeconomics is a branch of economics, which is the study of how customers and producers make decisions and how they interact in markets [10]. Customer preference is an important factor in making decisions of product sales, which thus becomes one major concern in micro economics. Kleinberg et al. [5] claimed that several micro economic problems can be solved by data mining techniques, which motivate the researchers in the database community to deal with the microeconomic problems. Due to different applications, the studies related to the microeconomic problems can be categorized into three types including the potential customers finding, the product advantages discovery, and the product positioning.

The majority of research [1], [4], [6], [8], [14], [15], [19] relevant to microeconomic problems has focused on the potential customers finding. This is to help a company find out the potential customers who may be interested in its specified product, and then the company can advertise the product to the potential customers. In this way, the company may gain profit only from the identified potential customers. To attract more customers' attention, the issue of the product advantages discovery addressed in [11] and[18] is to find the merits of the specified product of a company. The company can promote the product by using the found merits, and further advance the competitiveness of the company in the market. Nevertheless, the attention of the product advantages discovery is centred on an existing product whose characteristics have been known, and consequently the product may not satisfy the customers even though its merits are known. In recent years, new studies in [7], [16], and [20] appeared that tackled the issue of product positioning strategies. The purpose of the studies in this type is to help companies develop new products satisfying the needs of the customers within the target market, which is also the goal of our work.

Many studies have dealt with the potential customers finding, such as the reverse k-nearest neighbour query [1],[6],[14], [19], the reverse skyline query [4], [8], and the reverse top-k query [15]. The concepts of these works are similar. Given a set of customer preferences and a specified product, the queries studied in [1], [4], [6], [8], [14], [15], and [19] return the customers whose favourite products contain the specified product according to their customer preferences. The

visibility of the product is, therefore, limited to the potential customers.

On the other hand, the goals of the studies in [11] and [18] are to increase the visibility of the specified product by discovering the product advantages. The study in [18] aims at discovering features of a product by which the rank of the specified product is the top of all the products according to a given scoring function. Since it does not take customer requirements into consideration, consequently the customers may not be interested in the discovered product advantages. Considering the customer requirements, Miahet al. [11] propose an algorithm to choose k features of the specific product, which satisfy the maximum number of customers. Using the found merits to promote the product should have the higher opportunity to attract more customers' attention than the manner in the first type. Never the less, the works in this type focus on an existing product whose characteristics are fixed and it is possible that most customers are not interested in the product.

To help companies develop the products which are popular with the customers, the aim of the studies in [7],[16], and [20] is to determine the right positioning for products in the production plan. Given a set of existing products with multiple components, Wan et al. [16] consider the problem of producing better products than existing ones with cooperative companies. However, the customer requirements are not taken into consideration, which is one major factor in microeconomics. In addition, the number of the new products can be extremely large. As a result, the manager of the company may be over whelmed when he/she has to select several new products manually to identify the ones that will eventually be regarded as competitive with the existing products.

Li et al. [7] extend the concept of dominance, used in the Skyline operator [3], [9], [12], [13], [17], for business analysis. Given a set of customer requirements and the profit constraint of a company, the problem addressed in [7] is to identify the product dominating the largest customer requirements, which satisfies the profit constraint of the company. Extended from [7], suppose there are numerous companies with their respective profit constraints and a set of customer requirements, by taking competition into consideration, the goal of [20] is to find one product with the maximum expected number of the customers for each company, which satisfies the profit constraint of the company. In summary, the found product in [7] and [20] has to satisfy the profit constraint of the company, which may be difficult to specify. Moreover, to attract more customers, a company may choose to offer multiple products at the same time. The studies in [7] and [20] consider only one product for a company, and consequently cannot reflect the need in the real world. In this paper, taking both product competition and customer requirements into consideration, we propose methods to find k products from all candidate products a company can offer such that the expected number of the total customers for the k products is maximized.

3. Problem Statement

In this section, we formally define the k-MDP discovering problem and show that it is an NP-hard problem when the number of attributes for a product is larger than or equal to 3.

3.1 Formal Problem Definition

Assume that there is a nonempty set of customers, denoted C, demanding a certain type of products. Besides, EPde notes a set of existing products of the type. For each product, d numerical attributes, named quality describers, are used to represent the quality of the product in various aspects. Thus, the quality of an existing product ep in EP is represented by a vector <ep[1], ep[2] . . . , ep[d]>, where ep[i]denotes the value of ep on the it h quality describer. In addition, for a customer c in C, the requirements on the products are represented by a vector <c[1], c[2], . . . , c[d]>,describing the quality constraint on the value of each quality describer. Suppose that there is a nonempty set of candidate products, denoted CP, which can be offered by a company. The quality of a candidate product is also represented by a vector of the quality describers.

Without loss of generality, we assume that a smaller value of a quality describer indicates better quality in the corresponding aspect. Therefore, a product p, which is an existing product or a candidate product, is said to satisfy a customer c if and only if p[i]≤ c[i] for 1≤ i ≤ d. In addition, it is supposed that each customer c in C will definitely purchase one of the products which satisfy his/her requirements. If there is more than one product satisfying the requirements of c, the probability of these products purchased by c is assumed equal.

Suppose that there are a set c of customers, a set EP of existing products, and a set CP of candidate products which can be offered by a company, we want to help the company to select k products from CP such that the expected number of the total customers for the k products is maximized.

Let kCP denote a set of k products chosen from CP, cp denote a product in kCP, and c denote a customer in C. In addition, N (EP, c) and N (kCP, c) denote the total number of products in EP and kCP satisfying c, respectively. Since c will definitely purchase one of the products satisfying his/her requirements, if cp satisfies the requirements of c, the probability of c purchasing cp is inverse proportional to the total number of products in EP and kCP satisfying c, otherwise, it is 0. Consequently, the probability of a customer c purchasing a product cp in kCP, denoted P (cp, c) is calculated as follows:

$$\mathbf{P(cp, c) = \frac{1}{N(Ep, c) + N(kCP, c)}\ if\ cp\ satisfies\ c,}$$

P (cp, c) =0, otherwise (1)

Where the value of N (EP, c) + N (kCP, c) represents the total number of products in EP and kCP satisfying c. Accordingly, for a product cp in kCP, the expected number of the customers in C is obtained by adding the probabilities of each customer c in C purchasing cp as follows:

$$\mathbf{E(\{cp\}, C) = \sum_{c \in C} P(cp, c).\ (2)}$$

Furthermore, the expected number of the total customers in C for the set kCP, denoted E(kCP,C), is defined by adding the expected number of the customers in C for each product cp in kCP as follows:

$$\mathbf{E(kCP, C)} = \sum_{\mathbf{cp \in kCP}} \mathbf{E(\{cp\}, c)} = \sum_{\mathbf{cp \in kCP}} \sum_{\mathbf{c \in C}} \mathbf{P(cp, c)}. \quad \mathbf{(3)}$$

Definition 1 (k-MDP). *Given a set C of customers, a set EP of existing products, a set CP of candidate products, and a positive integer k less than |CP|, the k-MDP are the k products chosen from the set CP with the maximum E(kCP, C).*

Example 3.1. Suppose that there are nine customers, three existing rental properties, and seven candidate rental properties as shown in Fig. 1. There are two quality describers denoted by the distance to a school and to a subway station, respectively. Suppose that k is set to 3,any subset of {cp1, cp2, cp3, cp4, cp5, cp6, cp7} consisting of three elements forms a set of three candidate rental properties. For example, the expected number of the total customers in C for {cp2, cp3, cp4} is computed as follows:

$$E(\{cp_2, cp_3, cp_4\}, C) = \left(\frac{1}{1+2} + \frac{1}{0+1} + \frac{1}{3+3} + \frac{1}{3+3} + \frac{1}{3+3} + \frac{1}{3+3} + \frac{1}{2+3} + \frac{1}{3+3} + \frac{1}{3+3}\right) + \left(\frac{1}{1+2} + 0 + \frac{1}{3+3} + \frac{1}{3+3} + \frac{1}{3+3} + \frac{1}{3+3} + \frac{1}{2+3} + \frac{1}{3+3} + \frac{1}{3+3}\right) + \left(0 + 0 + \frac{1}{3+3} + \frac{1}{3+3} + \frac{1}{3+3} + \frac{1}{3+3} + \frac{1}{2+3} + \frac{1}{3+3} + \frac{1}{3+3}\right) = 5.26$$

Among all the possible sets, {cp2, cp3, cp4} gets the largest value of E (kCP, C). Consequently, the set {cp2, cp3, cp4} is the optimal solution of the 3-MDP discovering problem.

3.2 Computational Complexity

In this section, we show the k-MDP discovering problem is an NP-hard problem when the number of quality describers for a product, i.e., d, is larger than or equal to 3.

Theorem 1. *The k-MDP discovering problem is NP-hard when the number of quality describers for a product is larger than or equal to 3.*

Proof. According to the proof proposed in [9], the problem of top-k representative skyline points (top-k RSP), which finds k objects from a set of skyline objects with most dominated objects, is an NP-hard problem when the dimensionality is 3 or more.

Let us consider a special case of the k-MDP discovering problem. First, there is no existing product. Besides, the requirements of each customer are satisfied by at least one candidate product and no candidate product dominates each other. Thus, each candidate product corresponds to a skyline object and each customer requirement corresponds to a non skyline object.

In this special case of the k-MDP discovering problem, let kCP denote a set of k candidate products chosen from CP. Because there is no existing product in the special case, if the requirements of a customer are satisfied by any candidate product in kCP, the probability of the customer purchasing one candidate product in kCP is 1,other wise, it is 0. In other words, E (kCP, C) is equal to the number of customers satisfied by any candidate product in kCP. Since each candidate product corresponds to a skyline object, kCP corresponds to a set of k skyline objects, denoted M, in the top-k RSP problem. Moreover, each customer corresponds to a non skyline object in the top-k RSP problem. Consequently (kCP, C) is equal to the total number of non skyline objects dominated by the skyline objects in M.

On the other hand, each skyline object in the top-k RSP problem corresponds to a candidate product in CP in the k-MDP discovering problem. Therefore, a set M of k skyline objects correspond to a set kCP of k candidate products in CP. Moreover, each non skyline object in the top-k RSP problem corresponds to a customer in C in the k-MDP discovering problem. Since there is no existing product in this special case, the number of non skyline objects dominated by the skyline objects in M is equal to E (kCP, C).

It is shown that the top-k RSP problem defined in [9] can be reduced in polynomial time to the special case of the k-MDP discovering problem. Since the top-k RSP problem has been proved to be NP-hard, the k-MDP discovering problem is also NP-hard.

4. Algorithms for Optimal Solutions

Although the greedy algorithms provide efficient performance for getting a solution of the k-MDP discovering problem, they do not guarantee to find the optimal solution. The most intuitive method to get the optimal solution is to perform exhaustive search. In other words, the subsets of CP with k candidate products are enumerated. The optimal solution is the set of k candidate products with the highest expected number of the total customers in C. However, it is computationally infeasible to find the optimal solution of an NP-hard problem. Therefore, two algorithms, the A priori based algorithm and the upper-bound pruning algorithm, are designed based on providing some pruning strategies to reduce the search space of the optimal solution.

4.1 Apriori-Based (APR) Algorithm

Similar to the Apriori algorithm [2], the APR algorithm generates all the sets containing a single candidate product first. Let S denote a set of l candidate products, where $1 \leq l < k$. For any kCP which contains S, denoted kCP_S, the main idea of the APR algorithm is to estimate the upper and lower bounds of E (kCP_S, C). The bound values are used to prune the sets of l candidate products whose supersets are impossible becoming the optimal solution of the k-MDP discovering problem. In the next iteration, the remaining sets of l candidate products ($1 \leq l < k$) are combined to generate the sets of (l + 1) candidate products. The above process will repeat until the sets of k candidate products are generated to discover the k-MDP.

Before introducing the APR algorithm, a series of properties about getting E (kCP, C) are derived. First, the expected number of the total customers for a set kCP of k candidate products defined by (3) can be transformed into

$$\mathbf{E(kCP, C)} = \sum_{\mathbf{cp\in kCP}} \sum_{\mathbf{c\in C}} \mathbf{P(cp, c)} = \sum_{\mathbf{c\in C}} \sum_{\mathbf{cp\in kCP}} \mathbf{P(cp, c)}$$
$$= \sum_{\mathbf{c\in C}} \frac{\mathbf{N(kCP, c)}}{\mathbf{N(EP, c) + N(kCp, c)}} \quad \mathbf{(5)}$$

In (5), $\frac{N(kCP,c)}{N(EP,c)+N(kCp,c)}$ represents the probability of a customer c purchasing any product in kCP, which is denoted as P(kCP, c) in the following.

Let kCP1 and kCP2 denote two different sets of k candidate products. It can be proved that if the number of products in kCP1 satisfying a customer c is greater than or equal to the number of products in kCP2 satisfying c, the probability of c purchasing any product in kCP1 is greater than or equal to the probability of c purchasing any product in kCP2.

Lemma 1. *Let kCP1 and kCP2 denote two sets of k candidate products and c in C, where kCP1 ≠ kCP2. If N(kCP1, c)≥ N (kCP2, c), then P(kCP1, c)≥ P(kCP2, c).*

Proof. According to the property of a proper fraction, if a positive number is added into both numerator and denominator of the proper fraction, a larger proper fraction will be obtained.

It is known that N (kCP1, c) ≥ N (kCP2, c). Thus, (N (kCP1, c) - N (kCP2, c))≥0.

$$\mathrm{P(kCP1, c)} = \frac{\mathrm{N(kCP1, c)}}{\mathrm{N(Ep, c) + N(kCP1, c)}}$$

$$= \frac{\mathrm{N(kCP2, c) + (N(kCP1, c) - N(kCP2, c))}}{\mathrm{N(EP, c) + N(kCP2, c) + (N(kCP1, c) - N(kCP2, c))}}$$

$$\geq \frac{\mathrm{N(kCP2, c)}}{\mathrm{N(EP, c) + N(kCp2, c)}} = \mathrm{P(kCP2.\, c)}$$

In the case that kCP1 and kCP2 contain a common proper subset S, if the number of products in (kCP1 - S) satisfying a customer c is greater than or equal to the number of products in (kCP2 -S) satisfying c, it is implied that the probability of c purchasing any product in kCP1 is greater than or equal to the probability of c purchasing any product in kCP2.

Lemma 2. *Let kCP1 and kCP2 denote two sets of k candidate products, where kCP1 ≠ kCP2 and kCP1 ∩kCP2 ≠∅. Besides denotes a nonempty subset of (kCP1 \ kCP2): If N(kCP1 – S,c) ≥N(kCP2 – S, c), then P(kCP1,c) ≥P(kCP2, c).*

Proof. According to the definition,

N (kCP1, c) =N (kCP1 - S, c) + N(S, c) and
N (kCP2, c) =N (kCP2-S, c) + N(S, c):

In addition, it is known that N (kCP1 – S,c) ≥ N(kCP2 – S, c) =>N(kCP1, c) ≥N(kCP2,c)

According to **Lemma 1**, it is derived that P (kCP1,c) ≥ P (kCP2, c)

If each customer c in C satisfies N (kCP1 – S, c) ≥ N(kCP2 – S, c), the expected number of the total customers in C for kCP1 will be greater than or equal to the expected number of the total customers in C for kCP2.

Theorem 2. *Given two sets kCP1 and kCP2 of k candidate products, where kCP1 ≠ kCP2 and kCP1 ∩kCP2 ≠∅. Besides denotes a nonempty subset of (kCP1 ∩ kCP2). If N (kCP1 -S, c) ≥ N (kCP2 –S,c) for each customer c in C, then E(kCP1, C) –E(kCP2, C).*

Proof. If N (kCP1 –S, c) ≥ N(kCP2 – S,c) for each customer c in C, from **Lemma 2**, it is derived that P(kCP1, c) ≥P(kCP2, c)for each c in C.

Accordingly, $\sum_{c\in C} P(kCP1, c) \geq \sum_{c\in C} p(kCP2, c)$ and $E(kCP_1, C) \geq E(kCP_2, C)$ is proved

Let S denote a set of l candidate products, where1 ≤ l < k, and N(S, c) denote the number of candidate products in S satisfying customer c. Besides, N (CP, c) denotes the number of candidate products in CP satisfying customer c. It is supposed that N (CP, c) and N(S, c) are known. Let U denote the set of candidate products, whose cardinality is (k - l), which are inserted into S to form a set $kCP_S$of k candidate products. For any set kCP_Swhich contains S, the upper bound and the lower bound of N(kCP_S, c), denoted UB N(kCP_S, c) and LB N(kCP_S, c)respectively, are estimated according to N(CP, c) and N(S, c)as follows:

Upper bound of N (kCP_S, c). An upper bound of N (kCP_S, c) occurs when all the candidate products in U Satisfying c. Moreover, since $kCP_S$is a subset of CP, it is impossible that N (kCP_S, c) is larger than N (CP, c). Therefore, N (CP, c) is another upper bound of N(kCP_S, c). To get a tighter upper bound, UB N (kCP_S, c) is got as follows:

$$\mathbf{UB\_N(kCP_s, c = Min(N(s, c) + (k - l), N(CP, c))} \quad \mathbf{(6)}$$

Lower bound of N(kCP_S, c). Among the (|CP| _ |S|) Candidate products which are not in S, there are (N(CP, c) - N(S, c)) products satisfying customer c. In other words, there are ((|CP| -|S|) -(N(CP, c) - N(S, c))) candidate products which are not in S and do not satisfy customer c. Since the cardinality of U is (k –l), if ((|CP| _ |S|) – (N(CP, c) - N(S, c))) is larger than or equal to (k -l), a lower bound of N(kCP_S, c) occurs when none of the candidate products in U satisfy c. That is, LB N(kCP_S, c) = N(S, c). Otherwise, there are at least ((k - l) – ((|CP| - |S|) – (N(CP, c) –N(S, c))) products in U satisfying c. Therefore, LB_ N(kCPs, c) is got as follows:

$$\mathbf{LB_{N(kCP_s,c)} = Max(N(S, c), N(S, c) + \Big((k - l) - \big((|CP| - |S|) - (N(CP, c) - N(S, c))\big)\Big)} \quad \mathbf{(7)}$$

After getting the two bounds of N (kCP_S, c), according to Theorem 2, an upper bound and a lower bound of E (kCP_S, {c}), denoted UB_ E (kCP_S{c}) and LB _E (kCP_S,{c}), respectively, are computed by the following formulas:

$$\mathbf{UB\_E(kCP_s, \{c\}) = \frac{UB\_N(kCP_s,c)}{N(EP,c)+UB\_N(kCP_s,c)}} \quad \mathbf{(8)}$$

$$\mathbf{LB\_E(kCP_s, \{c\}) = \frac{LB\_N(kCP_s,c)}{N(EP,c)+LB\_N(kCP_s,c)}} \quad \mathbf{(9)}$$

Consequently, an upper bound and a lower bound of E (kCP_S, C), denoted UB _E (kCP_S, C) and LB_ E (kCP_S, C), respectively, are computed by the following formulas:

$$\mathbf{UB\_E(kCP_s, C) = \sum_{c \in C} UB\_E(kCP_s, \{c\})} \quad \mathbf{(10)}$$
$$\mathbf{LB\_E(kCP_s, C) = \sum_{c \in C} LB\_E(kCP_s, \{c\})} \quad \mathbf{(11)}$$

In the *l*th iteration of the APR algorithm ($1 \leq l < k$), the set of all the sets of l candidate products, denoted CPSl, is generated by combining the sets of (l – 1) candidate products remained in the previous iteration. Let S1 and S2 denote two sets of l candidate products in CPSl. If LB_E (kCPS2, C) is larger than UB_E (kCPS1, C), E (kCPS1, C) must be less than E (kCPS2, C). In other words, none of the sets of k candidate products containing S1 will become the optimal solution of the k-MDP discovering problem. Consequently, S1 can be pruned such that it is not necessary to generate the longer sets containing S1. The iterative process is repeated until the sets of k candidate products are generated. Finally, the k-MDP is discovered by selecting the set kCP of k candidate products with the highest E (kCP,C).

5. Conclusions and Future Work

In this paper, we formulate the k-MDP discovering problem for determining k most demanding products with the highest expected number of the total customers. We have showed that the problem is NP-hard when the number of quality describers for a product is 3 or more. Accordingly, two greedy algorithms, the SPG algorithm and the IG algorithm, are proposed to find the results approaching the optimal solution. Moreover, two effective pruning strategies are provided to develop two algorithms, the APR algorithm and the UBP algorithm, for attempting to find the optimal solution of the problem. The performance for all the proposed algorithms on efficiency is improved with the BMI index structure. The experiment results demonstrate that the execution time of all the proposed algorithms is much less than that of the exhaustive search approach. More specifically, the UBP algorithm is comparable to the two greedy algorithms not only on the efficiency but also on the memory utilization with a small k setting. In addition, the IG algorithm is scalable to a large value of k. Consequently it is a good alternative to the UBP algorithm when the value of k becomes large.

The probability of a product purchased by a customer may be influenced by the values of the quality describers of the product. In addition, in some applications, nominal attributes are used to describe the characteristics of a product in some aspects, whose orderings depend on the preferences of the users. How to extend our framework for these additional issues is under our investigation.

References

[1] E. Achtert, C. Bohm, P. Kroger, P. Kunath, A. Pryakhin, and M.Renz, "Efficient Reversek-Nearest Neighbor Search in ArbitraryMetric Spaces,"Proc. 25th ACM SIGMOD Int'l Conf. Management ofData,pp. 515-526, 2006.

[2] R. Agrawal and R. Srikant, "Fast Algorithms for MiningAssociation Rules in Large Databases,"Proc. 20th Int'l Conf. VeryLarge Data Bases,pp. 487-499, 1994.

[3] S. Borzsonyi, D. Kossmann, and K. Stocker, "The SkylineOperator,"Proc. 17th Int'l Conf. Data Eng.,pp. 421-430, 2001.

[4] E. Dellis and B. Seeger, "Efficient Computation of Reverse SkylineQueries,"Proc. 33rd Int'l Conf. Very Large Data Bases,pp. 291-302,2007.

[5] J. Kleinberg, C. Papadimitriou, and P. Raghavan, "A Microeco-nomic View of Data Mining," Data Mining and KnowledgeDiscovery,vol. 2, no. 4, pp. 311-322, 1998.

[6] F. Korn and S. Muthukrishnan, "Influence Sets Based on ReverseNearest Neighbor Queries,"Proc. 19th ACM SIGMOD Int'l Conf.Management of Data,pp. 201-212, 2000.

[7] C. Li, B.C. Ooi, A.K.H. Tung, and S. Wang, "DADA: A Data Cubefor Dominant Relationship Analysis," Proc. 25th ACM SIGMODInt'l Conf. Management of Data,pp. 659-670, 2006.

[8] X. Lian and L. Chen, "Monochromatic and Bichromatic ReverseSkyline Search over Uncertain Databases," Proc. 27th ACMSIGMOD Int'l Conf. Management of Data,pp. 213-226, 2008.

[9] X. Lin, Y. Yuan, Q. Zhang, and Y. Zhang, "Selecting Stars: The k Most Representative Skyline Operator,"Proc. 23rd Int'l Conf.Data Eng.,pp.86-95, 2007.

[10] N.G. Mankiw,Principles of Economics,fifth ed. South-WesternCollege Pub, 2008.

[11] M. Miah, G. Das, V. Hristidis, and H. Mannila, "Standing Out in aCrowd: Selecting Attributes for Maximum Visibility,"Proc. 24thInt'l Conf. Data Eng.,pp. 356-365, 2008.

[12] H.Z. Su, E.T. Wang, and A.L.P. Chen, "Continuous ProbabilisticSkyline Queries over Uncertain Data Streams," Proc. 21st Int'lConf. Database and Expert Systems Applications,pp. 105-121, 2010.

[13] K.-L. Tanz, P.-K. Eng, and B.C. Ooi, "Efficient Progressive SkylineComputation,"Proc. 27th Int'l Conf. Very Large Data Bases,pp. 301-310, 2001.

[14] Y. Tao, D. Papadias, and X. Lian, "ReversekNN Search inArbitrary Dimensionality,"Proc. 30th Int'l Conf. Very Large DataBases,pp. 744-755, 2004.

[15] A. Vlachou, C. Doulkeridis, Y. Kotidis, and K. Norvag, "ReverseTop-kQueries,"Proc. 26th Int'l Conf. Data Eng.,pp. 365-376, 2010.

[16] Q. Wan, R.C.-W. Wong, I.F. Ilyas, M.T. Ozsu, and Y. Peng,"Creating Competitive Products,"Proc. 35th Int'l Conf. Very LargeData Bases,pp. 898-909, 2009.

[17] W.C. Wang, E.T. Wang, and A.L.P. Chen, "Dynamic SkylinesConsidering Range Queries,"Proc. 16th Int'l Conf. Database Systemsfor Advanced Applications, 2011.

[18] T. Wu, D. Xin, Q. Mei, and J. Han, "Promotion Analysis in Multi-Dimensional Space,"Proc. 35th Int'l Conf. Very Large Data Bases,pp. 109-120, 2009.

[19] W. Wu, F. Yang, C.Y. Chan, and K.L. Tan, "FINCH: EvaluatingReversek-Nearest-Neighbor Queries on Location Data,"Proc. 34thInt'l Conf. Very Large Data Bases,pp. 1056-1067, 2008.

[20] Z. Zhang, L.V.S. Lakshmanan, and A.K.H. Tung, "On Domination Game Analysis for Microeconomic Data

Mining,"ACM Trans.Knowledge Discovery from Data,vol. 2, no. 4, pp. 18-44, 2009.

Author Profile

Kadiyam. Ramaiah obtained the B.Tech .degree in Information Technology (IT) from Chalapathi Institute of Engineering and Technology college ,Guntur. At present pursuing the M. Tech in Computer Science and Engineering (CSE) department at Guntur Engineering college, Guntur.

P. Anitha Rani obtained the B. Tech Degree from Sri C.R. Reddy Engineering College and M. Tech (CSE) from JNTU, Hyderabad. She has 10 years of teaching experience and working in Computer Science and Engineering (CSE) Department at Guntur Engineering College, Guntur.