International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 4 | ISSN : 2456-3307

Implementation of Data Mining Technique for Determining K-Most Demanding Products

[1]Advait Pundlik, [2]Vaibhav Hood, [3]Pawan Satpute, [4]Utkarsh Mandade, [5]Ankita Tripathi, [6]Prof. Kapil Hande

[12345]*BE Students, Department of Computer science and Engineering, Priyadarshini Bhagwati College of Engineering, Nagpur, Maharashtra, India*

[6]*Assistant Professor, Department of Computer science and Engineering, Priyadarshini Bhagwati College of Engineering, Nagpur, Maharashtra, India*

ABSTRACT

This paper formulates an issue for production design as k-most demanding products (k-MDP). Given an arrangement of clients demanding a specific kind of products with numerous traits, an arrangement of existing products of the sort, an arrangement of applicant products that organization can offer, and a positive whole number k, it causes the organization to choose k products from the hopeful products to such an extent that the normal number of the aggregate clients for the k products is augmented. One avaricious algorithm is utilized to discover surmised answer for the issue. Endeavour is likewise made to locate the ideal arrangement of the issue by evaluating the normal number for the aggregate clients of an arrangement of k competitor products for diminishing the pursuit space of the ideal arrangement.

Keywords : Algorithm for Knowledge Management, Data Mining, Decision Support, Performance Evaluation of Algorithm

I. INTRODUCTION

Microeconomics manages the clients and maker relationship and how they take decisions. The significant worry of microeconomics is the client inclinations which is a critical factor in settling on decisions identified with product deals. While making production arrangements or promoting systems identified with the product, organization needs to distinguish the product with greatest esteem or high utility (Chen 1999). This utility or qualities can be dealt with as the capacity of communication of the organization with alternate elements, for example, clients and contenders. Considering, ht e issue considered in this paper, is to locate the most elevated utility production gets ready for the organization where the utility of the production design is chosen is chosen by the normal number of aggregate clients for the product in design.

The issue can be characterized as, finding k-most demanding products (k-MDP), given arrangement of clients demanding particular sort of product with some characteristic, arrangement of existing product with same property, set of hopeful product that an organization can offer, at that point we can help organization to discover k applicant products to such an extent that for these k product the normal number of aggregate clients is expanded.

II. RELATED WORK

Kleinberg et al.[5] brought up that data mining strategy can be utilized to take care of the few issues identified with

Microeconomics. Microeconomics related issues can be separated into three distinct classes as potential client discovering, product advantage disclosure, and product situating. The issue of potential clients finding has been attempted to illuminate in paper [2, 5, 7, 9, 12, 13, and 16].

Reverse k-nearest neighbour query is used in paper [2, 3, 5, 6]:

In k-closest neighbour algorithm, we endeavour to discover all items in data set with k closest neighbour according to particular of question protest. In this paper a proficient algorithm has been introduced to discover k-closest neighbour.

The technique called turn around horizon question is utilized as a part of paper [7, 8]:

In invert horizon questions, multidimensional dataset P is considered to an inquiry point q. Here in data space point q moves toward becoming root and all purpose of P are spoken to by their separation vector to q. The switch horizon inquiry algorithm the items whose dynamic horizon contain question protest q.

In all these exploration papers input is client inclinations and product name and yield is clients whose most loved products contain the predefined product all to their inclinations.

Advancement of another system to discover products which are prevalent with clients by determining right situating according to production design is given in papers. Downside of this technique is in spite of the fact that they consider production design. They don't think about client inclinations. The issue of discovering top clients for one particular product has been fathomed in papers [9, 10]. Downside is it thinks about just a single product. They don't think about client inclinations.

The issue of discovering top clients for one particular product has been fathomed in paper [3]. Downside is it think about just a single product.

In this paper both product rivalry and client necessities are thought about to discover k products from the hopeful products that an organization can offer in order to boost the normal number of the aggregate clients for the k products.

III. IMPLEMENTATION

This area gives a thought regarding how the bitmap structure is utilized to keep up the fantastic data of the data in EP& C. At that point 2 voracious algorithms are proposed to take care of the k-MDP finding issue.

A. BMI Index Structure

For the pre-handling of the data gathered a bitmap record is created taking the qualities of the clients into thought. These property estimations are embedded by clients. Bitmap Index is gathering of client inclinations in double organization.

Bitmap is totally non-blocking and endeavours a bitmap structure to rapidly distinguish whether a point is an intriguing point or not. Each record is mapped into an m-bit vector, where m is the entirety of the quantity of unmistakable characteristic esteems over all measurements. Dissimilar to existing bitmap structures which are regularly a bitmap rendition of the whole database, our bitmap structure is a pre figured piece structure with more data.

Give S a chance to indicate a set containing a solitary hopeful product cp in CP. The SPG algorithm utilizes E(S, C), which processes the normal number of the clients in C for S, as the positioning capacity of the hopeful products. The hopeful products with the best k estimations of the positioning capacity are chosen to shape a rough arrangement of the k-MDP finding issue.

Algorithm 1: The SPG Algorithm

Input: N_vector (EP, C), set C of client necessities, set CP of applicant product, and the esteem k
Output: An arrangement of k competitor product

1. i=0;
2. For every competitor product cp in CP
3. {compute the fulfillment bit string of cp;
4. S={cp};
5. Register E(S,C);
6. i=i+1;
7. In the event that i<=k
8. Embed E(S,C) into the best k list
9. Else if E(S,C)> the littlest incentive in top-k list
10. Supplant the littlest in top-k list with E(S,C);}
11. Put the comparing applicant products in the best k rundown to set the kCP;
12. Return kCP;

Let S indicates a vacant set at first. In every emphasis, the IG algorithm chooses one of the unselected competitor products cp, which has the greatest E(S {cp}, C) esteem. In the wake of embedding's the chose applicant product into S, the estimations of the choice capacity for the unselected hopeful products are recomputed in the accompanying emphasis to choose the following chose competitor product. The above procedure proceeds until k applicant products have been chosen.

Algorithm 2: The IG Algorithm

Input: N_vector (EP, C), set C of customer requirements, set CP of candidate product, and the value k
Output: A set of k candidate product

1. For each candidate product cp in CP
2. Compute the satisfaction bot string of cp;
3. S= null;
4. While |S|<k
5. { max_E=0;
6. For each candidate product cp in CP
7. temp_S=S {cp};
8. Compute E(temp_S,C);
9. If E(temp_S,C)>max_E
10. {
11. max_P={cp};
12. 11.
13. Max_E= E(temp_S,C) }}
14. 12.
15. S=S
16. max_P;
17. 13.
18. CP=CP – max_P; }
19. Return kCP;

Similar to the Apriori algorithm, the APR algorithm generates all the sets containing a single candidate product first. Let S denote a set of l candidate products, where 1 < l < k. For any kCP which contains S, denoted kCPs, the main idea of the APR algorithm is to estimate the upper and lower bounds of E (kCPs, 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 < 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.

Algorithm 3: The ARP Algorithm

Input: N_vector (EP, C), set C of customer requirements, set CP of candidate product, and the value k
Output: A set of k candidate product

1. For each candidate product cp in CP
2. Compute the satisfaction bit string of cp;
3. CPS1=null;
4. For each candidate product cp in CP
5. CPS1=CPS1 {{cp}};
6. For (l=1; l<k; l++)
7. {MAX=0;
8. For each S in CPS1
9. {Compute UB_E(kCPs,C);
10. Compute LB_E(kCPs,C);
11. If(LB_E(kCPs,C)>MAX) MAX+LB_E(kCPs,C);}
12. CPSl = Candidatecheck(MAX,CPSl);
13. CPSl+1 = Apriori_Gen(CPSl); }
14. MAX=0;
15. For each S in CPSk
16. { Compute E(S,C);
17. If (E(S,C)>MAX)
18. kCP=S; }}
19. Return kCP;

Function CandidateCheck(MAX, CPSl)
1. For each S in CPSl
2. If (MAX > UB_E(kCPs,C))
3. CPSl=CPSl -{S};
4. Return CPSl;

Function Apriori_Gen(CPSl)
1. CPSl+1= CPSl CPSl;
2. For each S in CPSl
3. If ((any subset of S with size l) CPSl) CPSl+1= CPSl+1- {S};

Return CPSl+1;

The solution found by SPG algorithm approaches the optimal solution. Therefore, the UBP algorithm takes the solution found by the SPG algorithm as the baseline solution kCPb. Besides, for each set kCP of k candidate products, another method is proposed to efficiently estimate the upper bound of E (kCP,C). The value of E (kCP,C) is required to be computed only when the upper bound of E(kCP,C) is larger than E(kCPb, C). The baseline solution kCPb will be updated if a better solution is found. Therefore, the final result of kCPb is the optimal solution. Moreover, a well-designed method to decide the checking order of the sets of k candidate products is provided such that the checking process of the UBP algorithm can be performed efficiently.

Algorithm 4: The UBP Algorithm

Input: N_vector (EP, C), set C of customer requirements, set CP of candidate product, and the value k

Output: A set of k candidate product
1. For each candidate product cp in CP
2. {compute the satisfaction bit string of cp;
3. S= {cp};
4. Compute E(S,C); }
5. SL=<cp1’,cp2’,….cp|cp|’> // according to decreasing values of E(S,C)’

IJCSNS International Journal of Computer Science and Network Security, VOL.14 No.6, June 2014 21

6. kCPb={cp1’,cp2’,……cpk’};
7. base=E(kCPb,C);
8. kCP={cp1’,cp2’,…..cpk-1’,cpk+1’};
9. prune={cp|cp|-k+1’,cp|cp|-k+2’,….,cp|cp|};
10. While (true)
11. { compute UB_E2(kCP,C);
12. If UB_E2(kCP,C)>base
13. { compute E(kCP,C);
14. If E(kCP,C)>base
15. { base=E(kCP,C);
16. kCPb=kCP; }}
17. Else prune=kCP;
18. kCP=NextCandidateGen(SL,prune,k);
19. If kCP= null Break; }
20. Return kCPb;

Fuction NextCandidateGen(SL,prune,k)
1. While(true)
2. {kCP= the next set of candidate product according to t
3. If prune <r kCP continue;
4. Else break ;}

Return kCP;

IV. RESULT ANALYSIS

To evaluate efficiency & accuracy the algorithms were implemented in java with MySQL database. The Trip Advisor Dataset considered collected from UCI Machine Learning Repository.

Fig. 1 shows the graph for the memory required for algorithms to execute whereas Fig. 2 shows the graph for the time require for algorithms to execute.

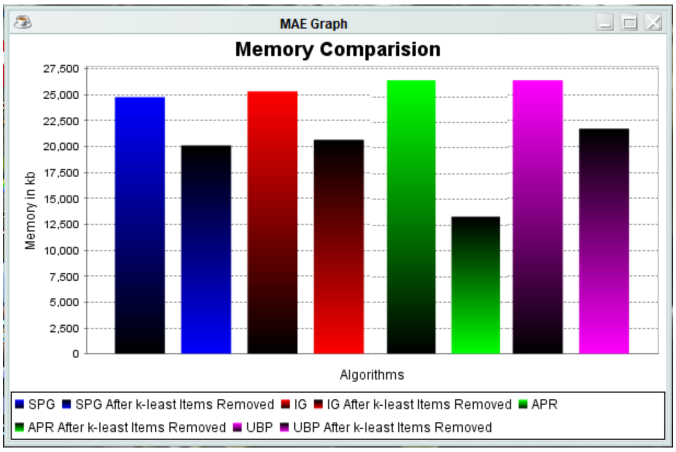

Figure 1 Memory Comparison

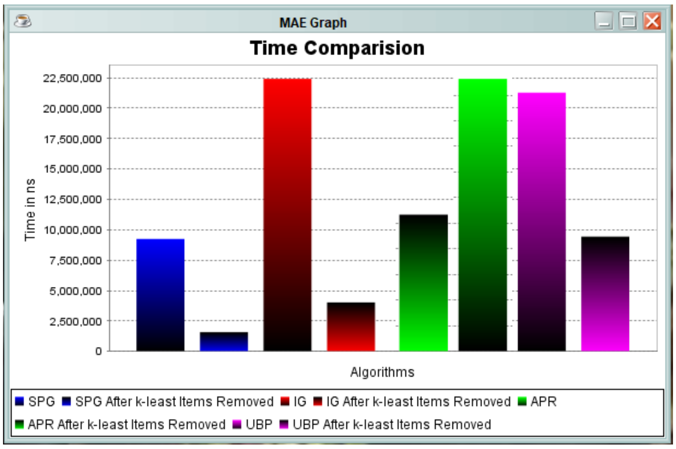

Figure 2 Time Comparison

We have analyzed the algorithms based on the experimental setup made. From the graph we can observe that the time required for ARP algorithm is maximum and the accuracy of the ARP algorithm is less. Whereas the accuracy of the UBP algorithm is more than any other algorithm & also the time required for this algorithm is minimum. So we can conclude that UBP algorithm is better than any other algorithm. Also we have observed that the all the algorithms execute properly for more than three attributes of the product.

V. CONCLUSIONS

In this paper k-most demanding product framework is actualized. It comprise of four phases in first stage the data is gathered from the enlistment of the clients, in second stage preprocessing is done on this data to create bitmap record, In next stage 4 distinct algorithms are executed to locate the best demanding products. Toward the end the algorithms are dissected to locate the better algorithm. The UBP has precision of 100% and time effectiveness as 0.678 sec. The ARP has precision of 33% and time productivity as 0.001 sec. So we can state that UBP algorithm is superior to some other algorithm.

VI. REFERENCES

[1]. Chen-Yi Lin, Jia-Ling Koh, and Arbee L.P. Chen, "Determining k-Most Demanding Products with Maximum Expected Number of Total Customers", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING.Harlow, England: Addison-Wesley, 2016.

[2]. J. Kleinberg, C. Papadimitriou, and P. Raghavan, "A Microeconomic View of Data Mining",Data Mining and Knowledge Discovery, vol. 2, no. 4, pp. 311-322, 1998.

[3]. E. Achtert, C. Bohm, P. Kroger, P. Kunath, A. Pryakhin, and M. Renz, "Efficient Reverse k-Nearest Neighbor Search in Arbitrary Metric Spaces",Proc. 25th ACM SIGMOD Intl Conf. Management of Data, pp. 515-526, 2006.

[4]. S. Borzsonyi, D. Kossmann, and K. Stocker, "The Skyline Operator", Proc. 17th Intl Conf. Data Eng., pp. 421-430, 2001.

[5]. Y. Tao, D. Papadias, and X. Lian,"Reverse kNN Search in Arbitrary Dimensionality", Proc. 30th Intl Conf. Very Large Data Bases, pp. 744-755, 2004.

[6]. W. Wu, F. Yang, C.Y. Chan, and K.L. Tan,"FINCH: Evaluating Reverse k-Nearest-Neighbor Queries on Location Data", Proc. 34th Intl Conf. Very Large Data Bases, pp. 1056-1067, 2008.

[7]. E. Dellis and B. Seeger, and K.L. Tan,"Efficient Computation of Reverse Skyline Queries", Proc. 33rd Intl Conf. Very Large Data Bases, pp. 291- 302, 2007.

[8]. X. Lian and L. Chen, and K.L. Tan,"Monochromatic and Bichromatic Reverse Skyline Search over Uncertain Databases",Proc. 27th ACM SIGMOD Intl Conf. Management of Data, pp. 213-226, 2008.

[9]. C. Li, B.C. Ooi, A.K.H. Tung, and S. Wang,"DADA: A Data Cube for Dominant Relationship Analysis", Proc. 25th ACM SIGMOD Intl Conf. Management of Data, pp. 659-670, 2006.

[10]. Q. Wan, R.C.-W. Wong, I.F. Ilyas, M.T. Ozsu, and Y. Peng,"Creating Competitive Products", Proc. 35th Intl Conf. Very Large Data Bases, pp. 898-909, 2009.