ijecs open access www.ijecs.in
International Journal Of Engineering And Computer Science ISSN: 2319-7242
Volume 4 Issue 9 Sep 2015, Page No. 14391-14395

Multi-criteria Decision making for Identifying Top-k profitable Stocks from Stock market

***Yara Srinivas*[#1], *Dr. A. Mary Sowjanya*[#2]**

#1 Department of Computer Science and Systems Engineering, A.U. College of Engineering(A), Andhra University, Visakhapatnam, Andhra Pradesh, India.yarasrinivas08@gmail.com
#2 Assistant Professor, Dept. of Computer Science and Systems Engineering, Andhra University, Visakhapatnam, Andhra Pradesh, India.sowmaa@gmail.com

Abstract:

In general, previous studies[1], [2]mostly focus on helping customers find a set of "best" possible products from a pool of given products. Finding top-k profitable products is common in many real-life applications like finding profitable laptops in a new laptop company, finding profitable delivery services in a new cargo delivery company, finding profitable shares in stock market and e-advertisements in a web page etc. This paper we propose a solution to a real life application by identifying top-k profitable stocks, since has not been studied before is undertaken. Given a set of stocks in the existing market, a set of k "best" possible stocks are found such that these new stocks are not dominated by the stocks in the existing stock market. Hence, the user can decide which stocks to be bought for making better profit. Two problem instances of finding top-k profitable stocks are found addressed in this paper. An extensive performance study using both synthetic and real data sets is reported to verify the effectiveness and efficiency of proposed algorithms.

Keywords: *Dominance and Skyline analysis, Skyline Query, Spatial database.*

I. Introduction:

Skyline:

Dominance and skyline analysis [1], [2] has been well recognized in multi-criteria decision-making applications. A package which is not dominated by any other packages is said to be a skyline package or it is in the skyline. The packages in the skyline are the best possible tradeoffs between the two factors in question.

The skyline operator [3] is important for several applications involving multi-criteria decision making. Skylines are related to several other well-known problems, top-K queries [4] and nearest neighbor search [5].

Example1: Multi-criteria Decision over Skyline.

Consider a table for names of hotels (Landmark, Tajmahal, Dolphin, Novotel, Royalport, GreenPark) with columns name, dist (distance to park), stars (quality rating), and price, as data shown in table1. This table has three metric columns dist, stars, and price. Based on these three metric dimensions, we could visualize the data as points in three-dimensional space.

The semantics of a Skyline query as shown in Query.1 is to find the maximal, through away any tuples that are dominated by others. The rows in black in the table.1 are the answers to the query. The rows that are bold out are those that were dominated. Landmark can eliminate, for example, by comparison with Green Park. Tajmahal is eliminated by comparison with Landmark, Dolphin or Green Park. Royal port is eliminated by comparison with Novotel. None of Dolphin, Novotel, or Green Park is dominated by any other. However, so these are in skyline and dominate other hotels.

```
select name, address
    from Hotel
    skyline of stars max,
             dist min,
             price min
```

Query 1: Sample Skyline Query

Name of hotel	Stars	Dist(km.s)	Price
LandMark	**	7	1175
Tajmahal	*	12	1237
Dolphin	*	**2**	**750**
Novotel	***	**2**	**2250**
Royal Port	***	5	2550
Green Park	**	**5**	**980**

Table 1: Sample Hotel Table

There may be many maximal solutions. A maximal need not be best on any one criterion. For example, Green park does not have the most stars is not the closest to the park and is not the least expensive. Rather, it represents a good balance of the criteria.

Given a set of points p_1, p_2…….p_n, the Skyline query returns a set of points P (referred to as the Skyline points), such that any point, $p_1 \epsilon P$ is not dominated by any other point in the data-set.

Skyline points:

A point p_i dominates another point p_j if and only if the coordinate of p_i on any axis is not larger than the corresponding coordinate of p_j.

Example 2: Skyline points

Data-set containing information about hotels, the distance to the beach and the price for each data point is recorded. Consider a two dimensional plot of the data-set, where the distance and price are assigned to the X,Y axis of the plot.

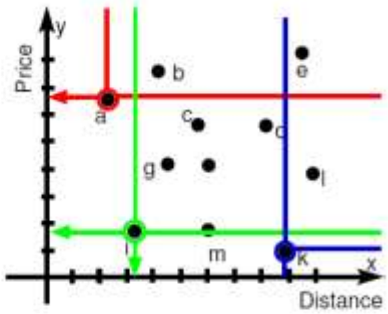

Figure 1: Example Skyline points

From above figure.1 the interesting data-points in the data-sets are {a, I, k}, 'a' has the least distance from the beach, k has the lowest price, I has neither the shortest distance nor the minimum price, 'I' has a lesser distance value than and a lower price value than 'a'. 'I' is hence not dominated by either one of them. All other points in the data-set are a dominated by the set of points {a, I, k}, i.e. both the distance and price values are greater than one or more Skyline points.

Definition for Top-k selection Query:

Consider a relation R, where each tuple in R has n attributes. Consider m scoring predicates, p_1………p_m define on these attributes.

Let $F(t)=F(p_1(t)\ldots\ldots p_m(t))$ be the overall score of tuple t ϵ R. A top-k selection query[4] selects the k tuples in R with the largest F value.

A SQL template for top-k selection query is the following:

SELECT *some _attributes*
FROM *R*
WHERE *selection_condition*
ORDER BY $F(P_1 \ldots . P_m)$
LIMIT k^1.

II. Existing system

The importance of dominance and skyline analysis has been well recognized in multi-criteria decision-making applications. Given a set of products in the existing market, we want to find a set of k "best" possible products such that these new products are not dominated by the products in the existing market. Most previous studies focus on how to help customers find a set of "best" possible products from a pool of given products. Problem of, finding top-k preferable products, which has not been studied before. Customer information systems such as travel agencies or mobile city guides are one application area for which Skyline queries are useful. Decision support (i.e., business intelligence) is another area.

For instance, Skyline queries [7], [8] can be used in order to determine customers who buy much and complain little. Furthermore, the Skyline operation is very useful for data visualization. With the help of the Skyline, the outline of a geometric object can be determined in other words, the points of a geometric object that are visible from a certain perspective can be determined using a Skyline query.

Another application is distributed query optimization, the set of interesting sites that are potentially useful to carry out a distributed query can be determined using a Skyline query. Those interesting sites have high computing power and are close to the data needed to execute the query.

- Skyline queries can also involve more than two dimensions and they could depend on the current position of a user. For instance, (mobile) users could be interested in restaurants that are near, cheap, and have good food (according to some rating system).
- The distance is based on the current location of the user. Again, the idea is to give the user the big picture of interesting options and then let the user make a decision. If the user moves on, the Skyline should be re-computed continuously in order to give the user a choice of interesting restaurants based on the user's new location.

III. Proposed system

Previous studies focus on how to help customers find a set of “best” possible products from a pool of given products. In this project, finding top-k profitable stocks along with their profit value [9]. This is the application of finding top-k products. Which has not been studied before is identified. Given a set of stocks in the existing stock market, a set of k “best” possible products are found such that these new stocks are not dominated by the stocks in the existing market. The main advantage of this project is to take lesser computational cost and directly can be applied for finding profitable as well as preferable i.e. popular stocks.

Two problem instances of finding top-k profitable stocks are found. In the first problem instance, price of stock is set such that the total profit is maximized and a utility function to be used to display the stock name along with their profit value. Which are not dominated by any other stocks in the existing market and such stocks are referred as top-k profitable Stocks.

In the second problem instance, k Stocks such that these k Stocks can attract the greatest number of customers and these makes customer decide what shares to be bought for making more profit. And also customer requirements into consideration, consequently the customers may not be interested in the discovered product advantages. Considering the customer requirements, in study propose an algorithm to choose *k* features of the specific Stocks, which satisfy the maximum number of customers. Such Stocks are referred as Top-k preferable Stocks.

In both problem instances, a straightforward solution is to enumerate all possible subsets of size k and find the subset which gives the greatest profit (for the first problem instance) or attracts the greatest number of customers (for the second problem instance). However, there are an exponential number of possible subsets.

In this paper, appropriate solutions are proposed and designed to find the top-k profitable Stocks and the top-k Preferable Stocks efficiently. An extensive performance study using both synthetic and real data sets is reported to verify the effectiveness and efficiency of proposed algorithms. The concept of the algorithms [1], [2] has been taken and modified according to our problem.

Finding Top-k System Structure:

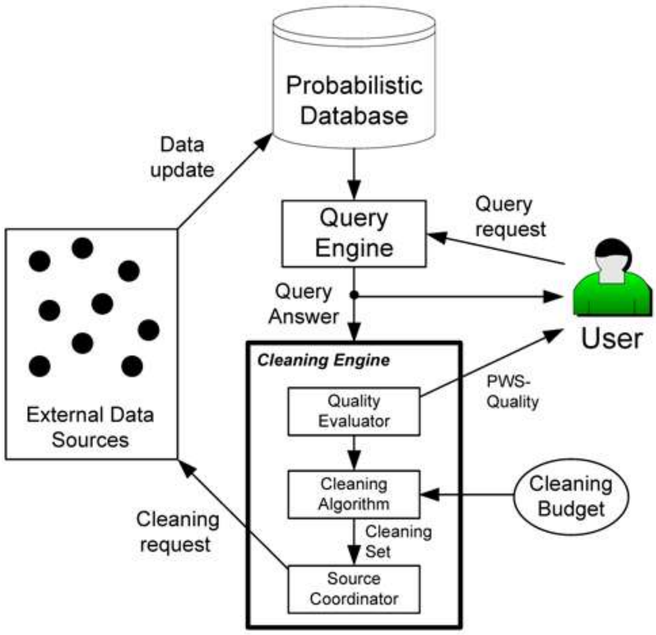

Figure 2: System Structure

DOI: 10.18535/ijecs/v4i9.56

Design and implementation of "Finding Top-k Profitable Products":

Finding top-k profitable products is common in many real-life applications. Other applications include finding profitable laptops in a new laptop company, finding profitable delivery services in a new cargo delivery company and finding profitable e-advertisements in a web page.

In this paper, how to find top-k profitable stocks from the market which are not dominated by existing stocks when data sets change. In some cases, data sets are dynamic and change from time to time. For example, some new products are launched in the existing market while some products which were present in the existing market become unavailable. Besides, the prices of existing products in the market may change due to various reasons, such as inflation and cost increase.

A naive way for this instance/problem is to enumerate all possible subsets of size k from Q, calculate the sum of the profits of each possible subset, and choose the subset with the greatest sum. However, this approach is not scalable because there are an exponential number of all possible subsets. This Project proposed efficient algorithms [1], [2] for problem Top-k profitable stocks using dominance analysis and skyline analysis. And also proposed dynamic programming approach which finds an optimal solution when there are two attributes to be considered. However, this problem is NP-hard [6] when there are more than two attributes to be considered. Thus, we followed two greedy algorithms for this problem. One greedy algorithm doesn't consider price correlation, while the other greedy algorithm considers price correlation.

For any two packages p and q, If p is better than q in at least one factor, and is not worse than q in any other factors, then p is said to dominate q. A package which is not dominated by any other packages is said to be a skyline package or it is in the skyline. Consider that a new travel agency wants to start some new packages from a pool of potential packages, and attribute price is to be determined by the

agency as its package price is not dominated by any package. In some cases, how we set the Price of a new package may affect how we set the price of another new package.

Given a set P of packages in the existing market and a set Q of potential new packages. In Finding Top-k Profitable Stocks (TPS), we want to select a set $Q^{\prime}$ of k packages from Q such that $F(Q^{\prime})$i.e., profit is maximized and each selected package is not dominated by any packages in the existing market and any selected new packages.

V. Results and Discussion:

The External source is (i.e. shares have to be either sold or purchased) added to Database. Data Administrate sends stocks to stock market. Now the customer registers his/her name and then logs into stock market database. The share portal will display top-5(according to algorithm) profit stocks along with their profit value in user account as shown in Fig.3 these are our required profitable and popular stocks. Now the user can decide which stocks to buy for making better profit in near future, and then log out from share portal (Stock market) user account.

Figure 3.Top-5 Profit Stocks

From above Figure.3, we observe that five stocks names along with their profit values are display in descending order (Mind tree 750, Infosys77, ranbaxy56, wipro45, GAIL32) in the share portal user account home page.

After the user operations are completed, entire data stored in data base along with their loss or profit details as show in Fig.4

DOI: 10.18535/ijecs/v4i9.56

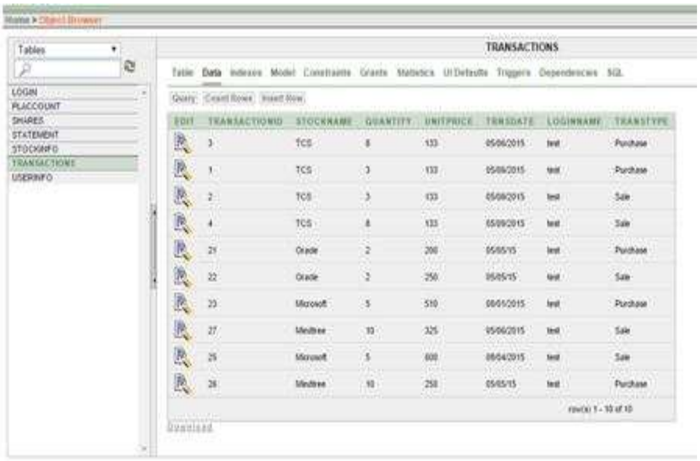

Figure 4. Stocks purchase or sell display in data base

VI. Conclusion

In this paper, how the costumer chooses the Top-k profitable as well as preferable stocks are not dominated by any other stocks existing in the market. This work proposes to choose the best stock to get maximum profit i.e. multiple decisions are arise when select the stock, even though it suggests best stock to get maximum profit and also all the problems for finding Top-k profitable stocks are solved and synthetic data has been used the result obtained are practically and theoretically accurate for testing. It is also suitable for real time data sets (i.e. companies real time trading records connected to server).

VII. Future Work

In the future, this work can be extended so as to work for specific applications of companies with their information and necessary permission. It can also be developed into a mobile app.

VIII. References:

1) Wong, Raymond Chi-Wing , Wan, Qian, “Finding Top-k Preferable Products” IEEE transactions on knowledge and data engineering, vol. 24, Issue: 10 Page(s): 1774 - 1788, Oct. 2012.

2) Q.Wan, R.C-W.Wong, and Y.Peng, “Finding Top-k profitable Products”, Proc. Int’l. Conf.Data minig (ICDE), 2011.

3) S. Borzsonyi, D. Kossmann,and K. Stocker,“The Skyline Operator,” Proc.Int’l Conf. Data Eng.(ICDE), 2001.

4) A. Vlachou, C. Doulkeridis, Y. Kotidis and K. Norvag, “Reverse Top-k Queries,” In Proceedings of the 26th International Conference on Data Engineering, pp. 365-376, 2010.

5) F. Korn and S. Muthukrishnan, “Influence Sets based on Reverse Nearest Neighbor Queries,” In Proceedings of the 19th ACM SIGMOD International Conference on Management of Data, pp. 201-212, 2000.

6) D.S. Hockhbaum, “Approximating Covering and Packing Problems: Set Cover, Vertex Cover, Independent Set, and Related Problems” Approximation Algorithms for NP-Hard Problems, PWS Publishing Company, 1997.

7) D. Papadias, Y. Tao, G. Fu, and B. Seeger, “Progressive Skyline Computation in Database Systems,” ACM Trans. Database Systems, vol. 30, no. 1, pp. 41-82, 2005

8) D. Papadias, Y. Tao, G. Fu, and B. Seeger, “An optimal and progressive algorithm for skyline queries,” in *Proc. ACM SIGMOD*, 2003, pp.467–478.

9) N. Archak, A. Ghose, and P.G. Ipeirotis, “Show Me the Money!: Deriving the Pricing Power of Product Features by Mining Consumer Reviews,” Proc. 13th ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining (KDD ’07), pp. 5665,2007.

10) Y. Tao, L. Ding, X. Lin, and J. Pei, “Distance-Based Representative Skyline,” ICDE ’09: Proc. IEEE Int’l Conf. Data Eng., pp. 892-903,2009

11) Q. Wan, R.C.-W. Wong, and Y. Peng, “Creating Top-K Profitable Products,” technical report, http://www.cse.ust.hk/~raywong/ paper/createTopKProfitableProducttech-nical.pdf, 2010.