ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

An Efficient Algorithm For Nearest Location Search Using Skyline Query Processing

Pooja Rao[1], Manglesh Khandelwal[2], Vinod Kumar Yadav[3]

M.Tech Scholar, Dept. of CSE, Bansal Institute of Research and Technology, RGPV, Bhopal (M.P.), India[1]

Assistant Professor, Dept. of CSE, Bansal Institute of Research and Technology, RGPV, Bhopal (M.P.), India[2]

Assistant Professor & HOD, Dept. of CSE, Bansal Institute of Research and Technology, RGPV, Bhopal (M.P.), India[3]

ABSTRACT: In this paper we proposed a technique for selecting Skyline Processing for nearest area every of that‘s near suited centers. Skyline operator is used in a query and carries out a filtering of result from a statistics base in order that it maintains best those items that aren’t worse than every other .This assist to manner to vicinity information in greater way and also improves consumer reaction time. In this paper we solve the hassle of KNN Query Processing in Entrusted Cloud Surroundings.

KEYWORDS: Skyline Query, Skyline Processing, Nearest Neighbor Search KNN.

I. INTRODUCTION

The skyline queries have gained lots of attention in various applications such as decision making. In such application skyline set retrieve the points that are not dominated by any other points. In database systems, queries specialized to search for the non-dominated data points are called skyline queries and their corresponding result set is known as skyline set [10]. Individual data points in a skyline results set are known as skyline points. The skyline queries which first addressed problem in static environment and it later progressively find out for dynamic set. If the user is start moving or query issued from a dynamic environment in that case it addresses the problem in dynamic environment. Location based service using skyline is one of popular application for skyline queries [8]. For example, in a passenger movement scenario it notifies their locations within frequent time interval to the passenger. So it helps to identify the best location which satisfies maximum user criteria. Besides all this advantage the main problem is skyline query processing and skyline result update are expensive application in database. The main cost is accessing data from storage and CPU cost spends for executing the user given query for dominance check. Search the efficiency and update criteria are the two most important performance criteria to skyline query processing and skyline result maintenance.

1.1 Skyline Query Processing

The skyline query processing has become an important issue in database research. The popularity of the skyline operator is mainly due to its applicability for decision making applications. In such application, the database tuples are represented as a set of multidimensional data points and the skyline set contains those points that are the best trade-offs between the different dimensions. Skyline queries aim to prune a search space of large numbers of multidimensional data items to a small set of interesting items by eliminating items that are dominated by others. Existing skyline algorithms assume that all dimensions are available for all data items. The main objective of skyline processing is minimizing query execution time. The Skyline operator [7] is used in a query and performs a filtering of results from a database so that it keeps only those objects that are not worse than any other. Definition: Given a set of points, the skyline query returns a set of points (referred to as the skyline points), such that any point is not dominated by any other point in the dataset. Definition of point domination: a point pi dominates another point if and only if the coordinate of pi on any axis is not larger than the corresponding coordinate of pj. Example: A dataset containing information about hotels; the distance to the beach and the price for each data point is recorded. Consider a two

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

dimensional plot of the dataset, where the distance and price are assigned to the X, Y axis of the plot. the goal of the search is to find a hotel whose distance to the beach and the price are both minimum (not restricted to minimum, any other function max, join, group-by clause could be used.) the preference function in our example is "minimum price and minimum distance". The dataset may not have one single data point that satisfies both these desirable properties. The user is presented with a set of interesting points that partly satisfy the imposed constraints.

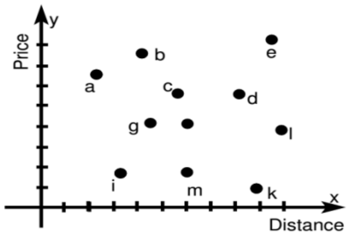

Figure 1: A 2D plot of dataset

1.2 Dynamic Skyline Query

Dynamic query means the query location is continuously changing so that skyline results are change frequently. The efficiency if skyline is computed in terms of accessing the data points and organizing the skyline result.

II. RELATED WORK

Many such cloud providers exist who offer powerful storage and computational infrastructures at low cost [1]. However, cloud providers are not fully trusted, and typically behave in honest-but-curious fashions. Specifically, they follow the protocol to answer the queries correctly, but they also collect the locations of the POIs and the subscribers for other purposes. Leakage of POI locations can lead to privacy breaches as well as financial losses to the data owners, for whom the POI dataset is an important source of revenue. Disclosure of user locations leads to privacy violations and may deter subscribers from using the service altogether. The skyline query is one very important query for users' decision making [2]. A skyline query returns a set of non-dominated data objects in a multi-dimensional dataset. Skyline query processing is multi-criterion data analysis tool which considers spatial attributes along with non-spatial attributes. A conventional approach includes both index based and non-index based approaches. A method to extend database systems has been suggested by using Skyline operations [3, 9]. This operation filters out a set of interesting points from a potentially large set of data points. A point is interesting if it is not dominated by any other point. For example, a hotel might be interesting for somebody traveling to Nassau if no other hotel is both cheaper and closer to the beach. In [4] an algorithm SFS has been developed, based on pre-sorting that is general, for use with any skyline query, efficient, and well behaved in a relational setting. The skyline, or Pareto, operator selects those tuples that are not dominated by any others. Extending relational systems with the skyline operator would offer a basis for handling preference queries. In [5] an algorithm to compute skyline has been developed which unlike most existing algorithms that compute the Skyline in a batch, returns the first results immediately, produces more and more results are continuously, and allows the user to give preference during the running time of the algorithm so that the user can control what kind of results are produced next.

III. PROPOSED ALGORITHM

A. *Design Considerations:*

The client has a query point Q and wishes to find the point's nearest neighbors. The query is encrypted using AES algorithm and sent to the server. The server receives the encrypted query and performs decryption using AES algorithm. The query is executed and K nearest neighbors is generated. The optimization is performed by

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

integration of skyline query parameters for filtering out the points which are not dominated by any other point. The system model comprises of three distinct entities:

- The data owner with the location data.
- The outsourced cloud service provider returning the processes information regarding the nearby locations.
- The client requesting for the nearest locations.

K nearest neighbors is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure (e.g., distance functions). KNN can be used for both classification and regression predictive problems. However, it is more widely used in classifications problems in the industry. To evaluate any techniques we generally look at 3 important aspects:

- Ease to interpret output
- Calculation time
- Predictive Power

B. *Description of the Proposed Algorithm:*

Nearest Neighbor Algorithm:

1. Retrieve all the dataset of a particular location using Google search API.
2. Filter out the nearest location x by finding the location(x_i, y_i)that is nearest to latitude longitude of the searched location according to Euclidean distance:

   $\|x - x_i\| = \sqrt{\sum_j (x_j - x_{ij})^2}$ ... (1)
3. Sort the dataset according to distance.

C. *Description of the proposed work:*

The large dataset obtained as search result consumes a lot of server resources and time which effects the client response time generated by server. Dynamic skyline query processing is added so as to efficiently reduce the resource consumption and memory requirements of server. Dynamic query means the query location is continuously changing so that skyline results are change frequently. The efficiency of skyline is computed in terms of accessing the data points and organizing the skyline result. In the improved approach utilize the Euclidean distance and treat closer objects as candidate skyline objects. It is assumed that the dominance relationship can be defined by the user as an additional category attribute as per the requirement like price, pin code etc. For efficient query processing, we employ the concept of nearest neighbor queries and evaluate progressively whether each object belongs to the skyline. If an object is not dominated by other objects in terms of the distance and all the category attributes, it belongs to the skyline. The proposed approach uses Dynamic query which means the query location is continuously changing so that skyline results are change frequently. The efficiency of skyline is computed in terms of accessing the data points and organizing the skyline result. A dynamic skyline query which report all points that are not dominated by any other points and it is calculated based on the distance between data point and query point. The current work uses filter and refine framework to handle the dynamic skyline query and during the filtering process it prune all the points that are dominated by some other points and generate set of candidate vertices by using BNL algorithm. A block-nested loop (BNL) is used to join two relations in a relational database. This algorithm is a variation on the simple nested loop join used to join two relations R and S.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

IV. SIMULATION RESULTS

A comprehensive evaluation on the proposed Skyline query is conducted. Experimental results demonstrate that the proposed algorithm is much faster than using a traditional Nearest Neighbor Search. Skyline also removes unwanted data from memory. So the memory space is been efficiently used.

Figure 2: CPU usage for processing data only with Nearest Neighbor Search.

The results were recorded on the basis of average time taken (figure 2) to perform transactions during Nearest Neighbor Search excluding skyline query processing. The graph shows average transaction response times for three virtual locations. The transactions took 8,000 seconds to perform (on average) when using the Durg to Bhopal virtual location.

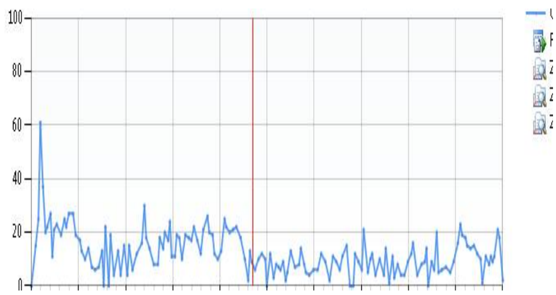

Figure 3: CPU usage for processing data Nearest Neighbor Search with Skyline processing

The Figure 3 shows that the approach is much better in terms of CPU usage which enables the usage of server resources in more efficient manner. The results had been recorded on the basis of average time taken to carry out transactions throughout nearest neighbor search together with skyline query processing. The graph shows average transaction response times for three virtual locations. The transactions took 8,000 seconds to perform (on common) when the usage of the Durg to Bhopal virtual area, the longest time for any of the three virtual places.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

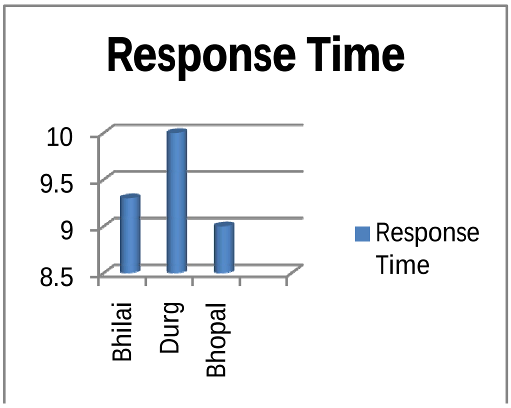

Figure 4: Response time excluding skyline query processing.

Again the results were recorded (Figure 4) on the basis of average time taken to perform transactions during Nearest Neighbor Search excluding skyline query processing. The graph shows average transaction response times for three virtual locations. The transactions took on an average 10.020 seconds to perform.

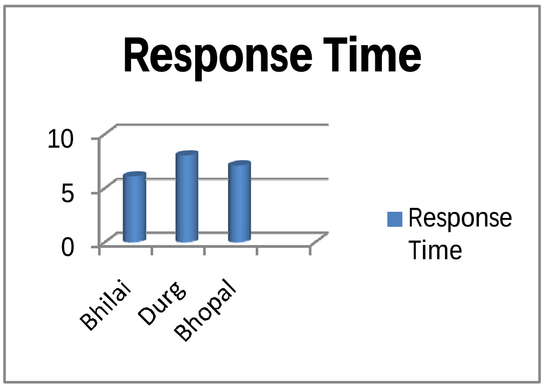

Figure 5: Response time including skyline query processing.

The above readings (figure 5) clearly show that the response time for skyline query processing performs is efficient and better. The graph shows average transaction response times for three virtual locations. The transactions took on an average 8.025 seconds to perform.

V. CONCLUSION AND FUTURE WORK

Recently skyline query processing which receives an interesting attention in data mining field. Skyline queries retrieves the non-dominated points from a large database system based on the user preference so it can be used in preference based applications. It successfully eliminates all dominated points by using some efficient technique. The current addresses some major problems of nearest location search like slow response time and heavy memory usage on servers integrating the dynamic skyline processing technique. Experimental results show that the approach outperforms the traditional Nearest Neighbor search in terms of resource usage and response time. Future work is required to develop an efficient caching of previous query results so as to reduce the computational cost for future queries.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 6, June 2017

REFERENCES

1. P.Rumao & P.Bhor , 'Secure kNN Query Processing in Untrusted Cloud Environments ', IJESC, Volume 6 Issue No.7,2016.
2. R. R. Meshram, 'Skyline Query Processing In Location Based Application', IJPRET, Volume 4 (9): pp. 861-866, 2016.
3. S. Borzsonyi, D. Kossmann, K.Stocker, 'The Skyline Operator', in: Proc. Int'l Conf. Data Eng, pp. 421-430, 2001.
4. J. Chomicki, Godfrey,J. Gryz , D. Liand ,' Skyline with Presorting', in: Proceedings of ICDE, pp. 717–816,2003.
5. D. Papadias, Y. Tao, G. Fu, B. Seeger,' Progressive skyline computation in database systems', in: ACM TODS 30(1), pp. 41–82, 2005.
6. K.L.Tan, P.K.Eng and B.C.Ooi, 'Efficient Progressive Skyline Computation', In: Proceedings of VLDB Conference, pp. 301–310, 2001.
7. X. Lin, J. Xu, H. Hu Range, 'Based Skyline Query in Mobile Environment', in: IEEE Transactions On Knowledge and Data Engineering, Volume. 25, no. 4, APRIL 2013.
8. Li, H., Tan, Q., Lee, W.,' Efficient progressive processing of skyline queries in peer-to-peer systems'. In: Proceedings of the International Conference on Scalable Information Systems (Infoscale), p. 26, 2006.
9. Y.Y Xiao, Y.G Chen, 'Efficient distributed skyline queries for mobile applications', in: Journal of Computer Science and Technology 25(3): pp. 523-536 May 2010.
10. H. Choi, H. Jung, Y.K Lee, and Y.D.Chung, 'Skyline queries on keyword-matched data', Information Science, pp.449-463, 2013.