Volume 3, Issue 8, August 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Secure and Dependable Cloud Storage and Retrieval for Metric Data

Sudheer Benarji.P* , **S.Sri Lavanya, R.V.Krishnaiah**

Department of CSE & JNTUH

India

Abstract - Cloud storage service is widely used to outsource data. However, the data owners have security concerns on the cloud storage services as the cloud service providers focus more on storage rather than security. This paper considers a scenario where data owners store and retrieve metric data into cloud. The data owners give access rights to their trusted clients. The trusted clients can make similarity queries on the outsourced metric data. This paper focuses on security mechanisms to be applied between the data owner and cloud server and trusted client and cloud server. Various techniques are built to transform and encrypt data before storing to database server for security reasons. The queries made by trusted clients are also protected in the same fashion. The resultant data is decrypted before sending to client. There were interesting trade offs observed in the mechanbisms. The empeircal results revealed that the mechanisms are capable of providing privay besides enabling accurate and efficient nearest neighbor (NN) queries.

Index Terms –Cloud storage, security, NN queries, metric data, cloud service provider

I. Introduction

Metric data is sensitive which comes from various fields like seismology, bioinformatics, medicine and astronomy.Since this data is increasingly becoming bulky, cloud computing has been considered as a source to store data with scalability and availability. Such data is very valuable to data owner for the reasons that he invests lot of money and time to collect such data. When confidential data such as metric data is outsourced to cloud straoge, naturally there are security concerns. Moreover, the cloud service providers focus more on storage facilities rather than security. Such data has to be stored securely in such a way that even the cloud service providers also should not be able to decrypt data. Since the cloud storage is affordable and viable, such data can be outsourced provided privacy guarantees.

The metric data is very sensitive. For instance NASA gets lot of such data through its programs such as Mars Express2, Moon1 etc. Such data is scientifically very valuable and rarely available data. Such data is considered to be private and confidential as they help NASA scientists to have valuable insights into space technologies. When such data is outsourced to cloud for storage services, the cloud is capable of providing scalable and available services. As far as security of such data is concerned the data owners are not sure about the guaranteed privacy of their data. When hackers are able to break the security, they can obtain valuable scientific data and get benefits out of them at the cost of NASA's essential trouble which can't be measured easily. The queries are related to distances and metrics. The data has to be stored consistency in order to get good results when clients make queries on metric data. Another example for metric data is DNA data which is analyzed by biologists. It contains the details and functioning of genes of human beings. For all genes experiments the data is collected and organized which makes it valuable. Such data when outsourced to cloud, it storage security assumes so much importance to its owner. Such databases are costly and very confidential for scientific purposes. This is the reason so much importance is attached to such data. Generating such data is also very costly and time consuming. When applications are made to make use of such data, privacy of the data while the data is at rest and also transit is essential. Queries can be made on the distances and similarity between the genes values. The values are closely related to various species in animal kingdom. There are many such scenarios where metric data is generated which can be outsourced to cloud.

This paper presents various security mechanisms to be used by data owner and also the data users or clients who make queries on data. The algorithms are tested with metric data. The experimental results revealed that the proposed mechanisms can guarantee the privacy of outsourced metric data. The remainder of this paper is structured as follows. Section II review relevant literature. Section III provides information about the architecture and security mechanisms of the proposed cloud storage system. Section IV provides experimental results while section V concludes the paper.

II. Prior Work

This section reviews the literature which is related to cloud data security, indexing NN search and metric space, privacy and security. Metric indexing is used by the proposed algorithms in order to make NN queries faster. Disk based indexes which are famous are X-tree [1] and R*tree [2]. These indexes are made for multidimensional objects such as time series objects or DNS sequences etc. Complex data objects can be presented and queries easily by having index. There are various surveys

made on metric data indexing. For instance it is elaborated in [3] and [4]. Mainly three representative indexing methods are described here. They include M-tree [5], MVP-tree [6] and vantage point tree (VP-tree) [7]. Out of all these things very famous one is M-tree. Its variant is also there as described in [8]. In the indexing of M-tree, each index entry consists of an anchor object, a covering radious, a point to child node and a precomputed distance from child to its parent. NN search process is important in searching metric data. The state-of-the-art algortim being widely used for NN queries is "best-first" approach [4], [9]. The algorithm computes the minimm distance between the given query and other objects in the search space. There are some other approaches that can be used for making queires efficient. They are known as hashing techniques [10], [11]. However, these techniques work well and retrieve the results from metric data set but without accuracy in results. For making such searches in multidimensional space Locality Sensitive Hashing (LSH) [11] technique is used. There is an extension to this technique for improveing its performance further is distance based hashing technique (DBH)[10] which take two parameters as input. They include number of C of hash tables and number of A of bits. The projection function of DBH as described in [12] is as follows.

$$PJF\ a_i b_i(p) = \frac{dist2(p,ai) + dist2(ai,bi) - dist2(p,bi)}{2.dist(ai,bi)}$$

In spite of its capabilities, the DBH algorithm has two known drawbacks. They are the possibility of no hash table having given object which leads to empty results and inability to optimize the query performance of DBH once it is built. These two problems are overcome in this paper by using a flexible hashing technique which prevents empty results besides allowing the user to improve the query performance by setting the tradeoff with respect to communication cost.

With regard to privacy and security the history of outsourcing has to be traced back before making some points clear. The first idea of outsourcing data is described in [13]. Encrypted solution with data confidentiality is provided for such data for the first time by [14]. It was achievedby mapping the records of a relational table into buckets and then gets encrypted records before storing to the server. Based on the description of the buckets, at query time, user finds the necessary buckets to get from server. The concept of data owner applying encryption is presented in [15]. In yet another kind of solution order-preserving function is employed in order to distribute values in different format [16]. The works which are related to our work in this paper include [17] and [18]. R-tree is used in the solution besides using the spatial transformations for security reasons. Outsourcing multidimensional data to storage services is presented in [18]. Secure scalar encryption production is used in this approach to ensure the data outsourced to cloud is protected and privacy is guaranteed. However, the drawback of this approach is that it does not use indexing concept. Therefore it lags behind proving query performance though it is capable of providing accurate results. It does mean that there is tradeoff between the query correctness and query efficiency. K-anonymity also has been applied to many applications where privacy preserving concept is required [19]. The proposed system tries to achieve both of them by developing various algorithms that are used by data owner, trusted clients and cloud data server. The proposed system ensures data privacy and security while providing accurate search results with good processing power.

III. Proposed Secruity Mechanisms

The proposed security mechanisms are going to solve the problems of cloud storage security to the outsourced metric data. The proposed solution to the problem overview has three entities namely server, data owner and trusted client. The operations of these three entities are visualized in fig. 1.

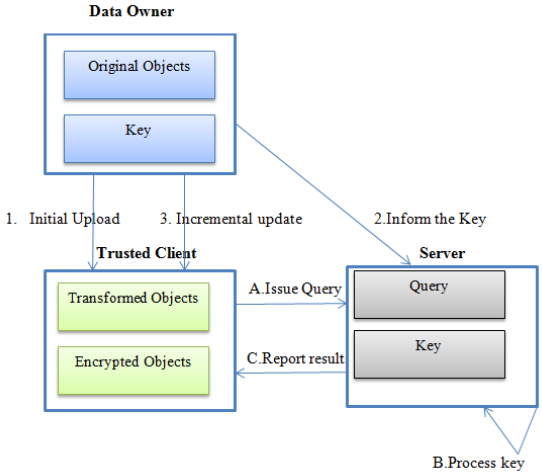

Fig. 1 –Overview of the proposed scenario

As can be seen in fig. 1, the data owner outsources metric data to cloud service provider. However, he does not store the plain text into server which is considered in this scenario untrusted. The data owner transforms data into some encoded format and sends the transformed objects to server. In the server indexing is also stored for those objects for faster processing of queries. Then the transformed objects are encrypted and finally stored in the database server. After uploading data to server securely, the data owner can make incremental update to his data. After storing data in cloud server, the data owner informs security key to his trusted clients also so as to enable them to make queries on the data and get the results to be decrypted. The trusted client queries to server and takes the results back from the server. The client makes transformed query to the server. The query results returned by the server will be decrypted by the trusted client.

Security Mechanisms

Towards securing outsourced metric data, three transformation functions have been proposed. These functions are responsible to convert the data objects uploaded by the data owner to another form without disturbing the metrics in the underlying data. Afterwards the encrypted data is stored in cloud server. The mechanism for data transformation is known as Encrypted Hierarchical Index Search, Metric Preserving Transformation, and Flexible Distance-Based Hashing. The first algorithm is used by client for making NN search on the outsourced data. The search is made on the encrypted index which is stored in cloud storage server. It provides perfect privacy to data.

```
1: request the server for the (encrypted) root node Loot;
2: H:=new min-heap; pnn:=NULL;
3: ϒ:=min_{e2Lroot}maxdist(q, e); . derive NN distance bound
4: for each entry e ϵ L_root such that mindist(q; e) < ϒ do
5: insert the entry (e, mindist(q, e)) into H;
6: while H is not empty and its top entry's key < ϒ do
7: pop next λ entries from H and insert them into a set S;
8: request the server for each (encrypted) child node of S;
9: for each retrieved node L_cur do
10: if L_cur is a leaf node then .check for closerobjects
11: update ϒ and p_nn by using objects in L_cur;
12: else . expand the entries of L_cur
13: ϒ:=minf{ϒ; min_{e2Lcur}maxdist(q, e)};
14: for each e ϵ Lcur such that mindist(q; e) }<ϒ do
15: insert the entry (e; mindist(q; e)) into H;
16: return p_nn as the result;
```

Listing 1 –EHI algorithm for searching

As can be seen in listing -1, the EHI algorithm is presented. This algorithm is meant for making NN search on the metric data present in cloud server. It has optimal data transfer cost while incurring large number of round trips to server. Listing 2 presents MPT building algorithm for data owner.

```
1:use a heuristic to select a set of A anchor
objects from P;
2: Integer B:=[|P|/A];
3: use a heuristic to assign each data object of
P to an anchor object, subject to the capacity
constraint B;
4: for i:=1 to A do
5: let a_i be the i-th anchor object;
6: let a_i:S be the set of objects assigned to the anchor a_i;
7: r_i:=max_{pai}:Sdist(ai, p); compute covering radius
8: for each object p ϵ a_i:S do
9: send the tuple {p:id;OPE(dist(a_i,p)) ECR(p; CK)} to the server;
```

Listing 2- MPT building algorithm

As can be seen in listing 2, MPT building algorithm is presented. It is used by data owner while sending or outsourcing data to server. Listing 3 presents the FDH building algorithm for data owner.

```
1: for i := 1 to A do . key generation
2: choose an object randomly from P as an anchor object a_i;
3: find the distance value r_i such that half of objects P ϵ P satisfy dist(a_i; p) < r_i;
4: for each object P ϵ P do
5: compute the encryption ECR(p;CK);
6: compute BM(p);
7: send the tuple (p:id;BM(p); ECR(p; CK)) to the server;
```

Listing 3 –FDH Building Algorithm

The FDH building algorithm presented in listing 3 finishes the communication with the server with just a single round trip. However, it does not make any guarantee for accuracy of results.

IV. Experiemntal Results and Evaluation

The environment used for developing a prototype web application used by data owner and trusted client is a PC with 2GM of RAM and core 2 dual processor. The software used is Visual Studio with ASP.NET technology and Visual C# programming language. Datasets used areYEAST, MUSH, SHUTL and GFC. Experiments are made using the dataset with all security mechanisms described in the previous section for both data owner and also trusted client. The construction time and server CPU time (seconds) required by the three algorithms is provided in table 1.

Table 1 –Construction and Server CPU Time for Three Transformations

Dataset	Construction Time			Server CPU Time		
	EHI	MPT	FDH	EHI	MPT	FDH
YEAST	0.016	0.094	0.313	0.001	0.001	0.049
MUSH	0.234	0.531	1.344	0.006	0.002	0.083
SHUTL	2.438	1.187	4.672	0.010	0.006	0.097
GFC	12.141	3.063	10.078	0.007	0.005	0.141

As can be seen in table 1, the construction time and server CPU time are presented. For YEAST and MUSH datasets, EHI has good performance both for construction time and also CPU time in server. For SHUTL dataset MPT has good performance in terms of construction time and CPU time in server. In case of GFC dataset, also MPT has better performance.

The evaluation of the algorithms with different datasets is visualized in the graphs presented in the following figures.

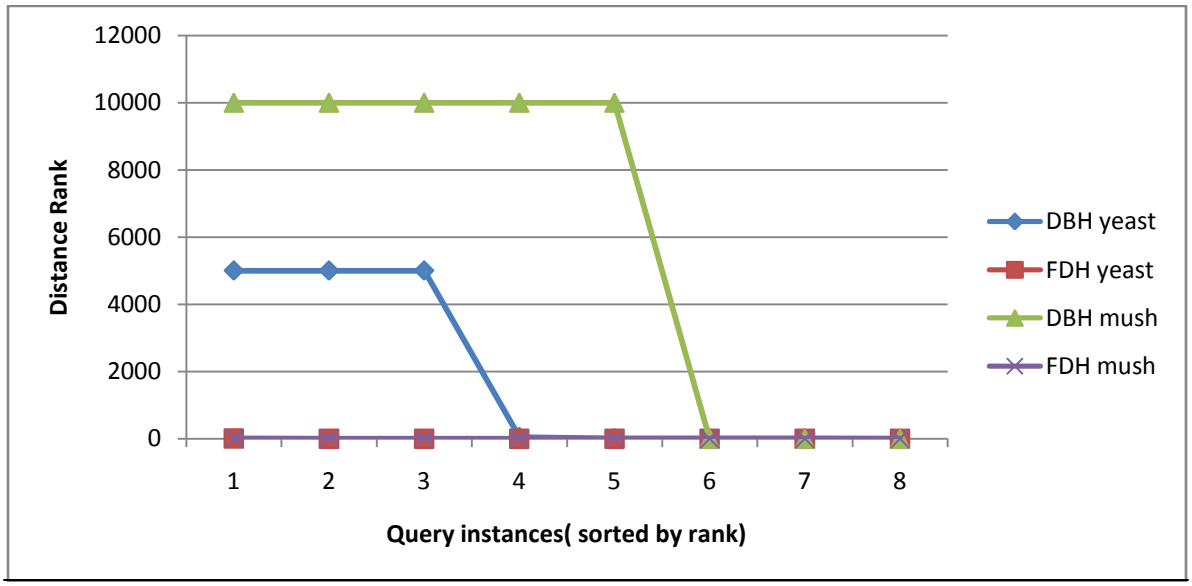

Fig. 2 –Rank of NN Search Results on YEAST and MUSH data

As can be seen in fig. 2, it is evident that the ranks of NN search results of DBH and FDB algorithms are presented for YEAST and MUSH datasets. On the datasets YEAST and MUSH the result rank of FDH is far better than that of DBH.

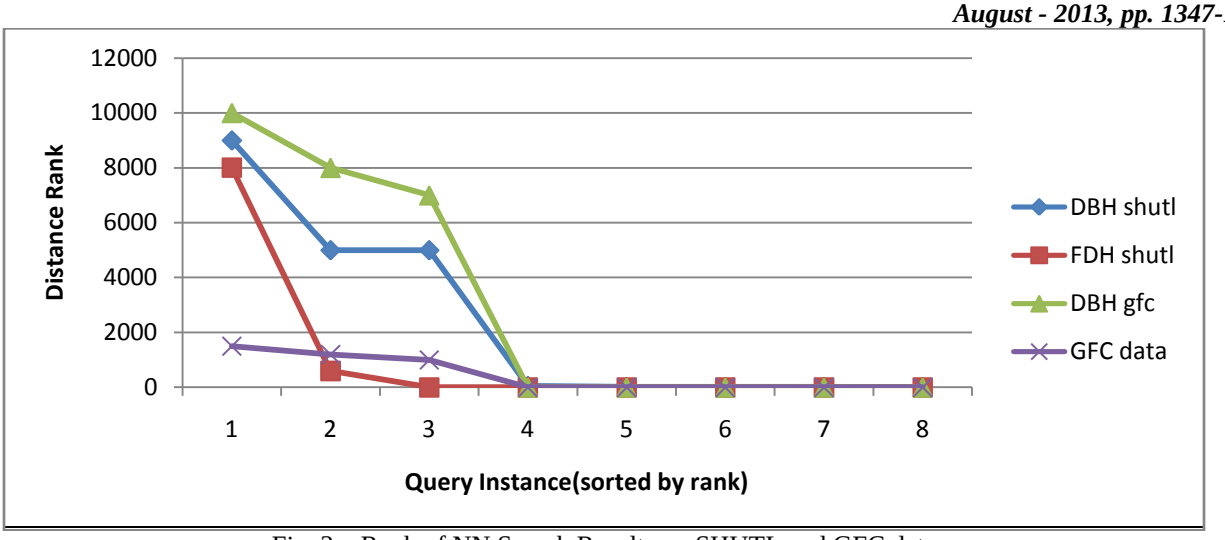

Fig. 3 – Rank of NN Search Results on SHUTL and GFC data

As can be seen in fig. 3, it is evident that the ranks of NN search results of DBH and FDB algorithms are presented for SHUTL and GFC datasets.

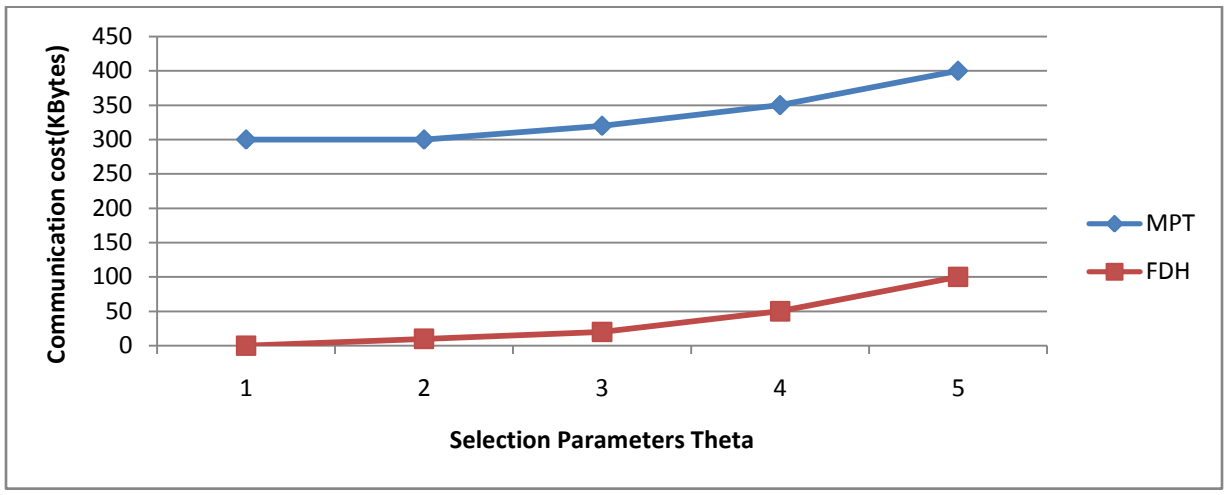

Fig. 4– Communication Cost

As can be seen in fig. 4 the communication cost of MPT and FDH algorithms is presented. As per the results shown the communication cost of FDH is far lesser than that of MPT.

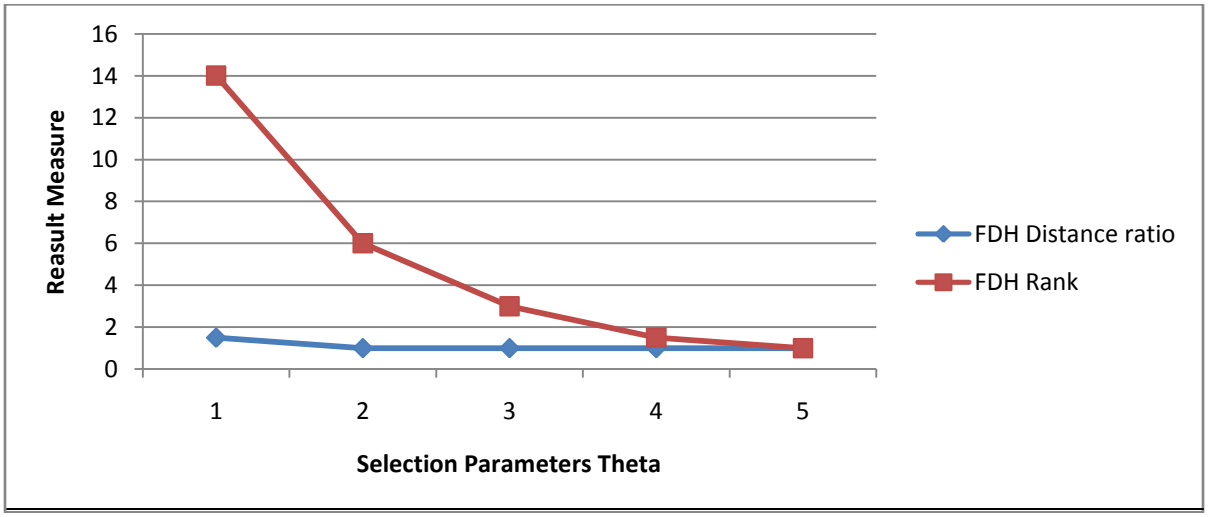

Fig. 5 –Result Measure

As can be seen in fig. 5, the result measure is presented with respect to FDH rank and FDH distance ratio.

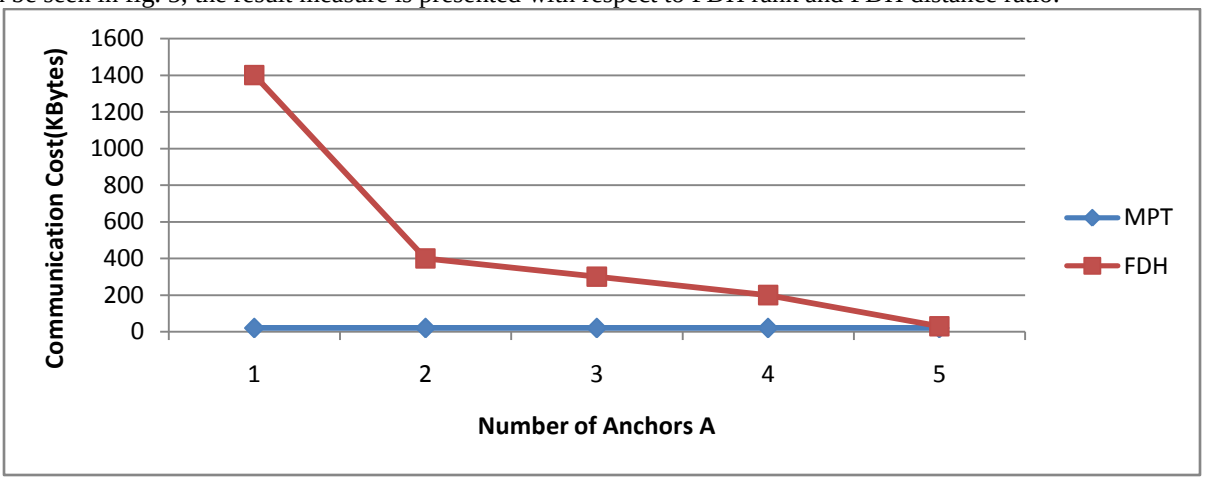

Fig. 6 – Communication Cost

As can be seen in fig. 6 the horizontal axis represents number of anchors while the vertical axis represents communication cost. They represent those details for both FDH and MPT algorithms. The MPT algorithm performance is far better that that of MPT with respect to number of anchors.

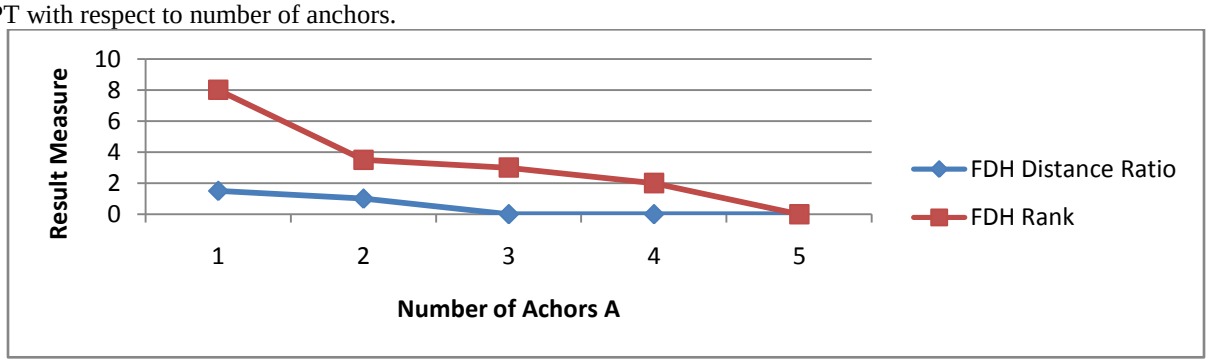

Fig. 7 – Result Measure

As can be seen in fig. 7, the result measure is presented with respect to FDH rank and FDH distance ratio with respect to number of objects.

V. Conclusion

We proposed various Nearest Neighbor (NN) search techniques that act on outsourced metric data such as metric data (bioinformatics) which is sensitive in nature. There are many existing solutions for NN search on metric data. However, they have got tradeoff between query efficiency and data privacy. In this paper we introduced search techniques that render both query efficiency and also data privacy. The search mechanisms are carried out by server. The Metric Preserving Transformation (MPT) is capable of storing distance data in server in the form of a set of anchor objects. However, it needs two rounds trips to the server for the task. The Flexible Distance based Hashing (FDH) reduces it to single trip. However, it does not give guarantees with respect to accuracy of results. Therefore these two are improved in order to ensure privacy and also efficiency. The experimental results revealed that the algorithms are efficient.

References

[1] S. Berchtold, D.A. Keim, and H.-P.Kriegel, "The X-Tree : An IndexStructure for High-Dimensional Data," Proc. 22nd Int'l Conf. VeryLarge Databases, pp. 28-39, 1996.

[2] N. Beckmann, H.-P.Kriegel, R. Schneider, and B. Seeger, "The R*-Tree: An Efficient and Robust Access Method for Points andRectangles," Proc. ACM SIGMOD Int'l Conf. Management of Data,pp. 322-331, 1990.

[3] E. Cha´vez, G. Navarro, R.A. Baeza-Yates, and J.L. Marroquı´n,"Searching in Metric Spaces," ACM Computing Surveys, vol. 33,no. 3, pp. 273-321, 2001.

[4] G.R. Hjaltason and H. Samet, "Index-Driven Similarity Search inMetric Spaces," ACM Trans. Database Systems, vol. 28, no. 4,pp. 517-580, 2003.

[5] P. Ciaccia, M. Patella, and P. Zezula, "M-Tree: An Efficient Access Method for Similarity Search in Metric Spaces," Proc. Very LargeDatabases (VLDB), pp. 426-435, 1997.

[6] T. Bozkaya and Z.M. O ¨ zsoyoglu, "Indexing Large Metric Spacesfor Similarity Search Queries," ACM Trans. Database Systems,vol. 24, no. 3, pp. 361-404, 1999.
[7] P. Yianilos, "Data Structures and Algorithms for NearestNeighbor Search in General Metric Spaces," Proc. Fourth Ann.ACM-SIAM Symp.Discrete Algorithms (SODA), pp. 311-321, 1993.
[8] C.T. Jr, A.J.M. Traina, B. Seeger, and C. Faloutsos, "Slim-Trees: High Performance Metric Trees Minimizing Overlap betweenNodes," Proc. Seventh Int'l Conf. Extending Database Technology(EDBT), pp. 51-65, 2000.
[9] T. Seidl and H.P. Kriegel, "Optimal Multi-Step k-NearestNeighbor Search," Proc. ACM SIGMOD Int'l Conf. Management ofData, pp. 154-165, 1998.
[10] V. Athitsos, M. Potamias, P. Papapetrou, and G. Kollios, "NearestNeighbor Retrieval Using Distance-Based Hashing," Proc. IEEE24th Int'l Conf. Data Eng. (ICDE), pp. 327-336, 2008.
[11] A. Gionis, P. Indyk, and R. Motwani, "Similarity Search in HighDimensions via Hashing," Proc. 25th Int'l Conf. Very LargeDatabases (VLDB), pp. 518-529, 1999.
[12] C. Faloutsos and K.-I. Lin, "FastMap: A Fast Algorithm forIndexing, Data-Mining and Visualization of Traditional andMultimedia Data Sets," Proc. ACM SIGMOD Int'l Conf. Managementof Data, pp. 163-174, 1995.
[13] H. Hacigu¨mu¨ s, S. Mehrotra, and B.R. Iyer, "Providing Database asa Service," Proc. 18th Int'l Conf. Data Eng. (ICDE), pp. 29-40, 2002.
[14] H. Hacigu¨mu¨ s, B.R. Iyer, C. Li, and S. Mehrotra, "Executing SQLover Encrypted Data in the Database-Service-Provider Model,"Proc. ACM SIGMOD Int'l Conf. Management of Data, pp. 216-227,2002.
[15] E. Damiani, S.D.C. Vimercati, S. Jajodia, S. Paraboschi, and P.Samarati, "Balancing Confidentiality and Efficiency in UntrustedRelational DBMSs," Proc. 10th ACM Conf. Computer and Comm.Security (CCS), pp. 93-102, 2003.
[16] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, "Order-PreservingEncryption for Numeric Data," Proc. ACM SIGMOD Int'l Conf.Management of Data, pp. 563-574, 2004.
[17] M.L. Yiu, G. Ghinita, C.S. Jensen, and P. Kalnis, "Outsourcing Search Services on Private Spatial Data," Proc. IEEE 25th Int'l Conf. Data Eng. (ICDE), pp. 1140-1143, 2009.
[18] W.K. Wong, D.W. Cheung, B. Kao, and N. Mamoulis, "Secure k- NN Computation on Encrypted Databases," Proc. 35th ACM SIGMOD Int'l Conf. Management of Data, pp. 139-152, 2009.
[19] L. Sweeney, "k-Anonymity: A Model for Protecting Privacy," Int'l J. Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 10, no. 5, pp. 557-570, 2002.

AUTHORS

Sudheer Benarji is student of DRK College of Engineering and Technology, Hyderabad, AP, INDIA. He has received B.Tech Degree computer science and engineering from VNRVJIET. M.Tech Degree in computer science and engineering. His main research interest includes Data Mining, Databases and DWH.

Sri Lavanya Sajja is working as an Assistant Professor in DRK College of Engineering and Technology, JNTUH, Hyderabad, Andhra Pradesh, India. She has received M.Tech degree in Computer Science from JNTUH along with an M.Tech degree in IT. Her main research interest includes Data Mining and Networking

Dr.R.V.Krishnaiah (Ph.D) is working as Principal at DRK INSTITUTE OF SCIENCE & TECHNOLOGY, Hyderabad, AP, INDIA. He has received M.Tech Degree EIE and CSE. His main research interest includes Data Mining, Software Engineering.

