PROVISION OF SECURITY TO UNSECURED OUTSOURCED DATA USING PROXIMITY MEASURE

Satyasri naga subbalaxmi[1], Mr. A. Krishna mohan[2]
1.P.G Student, M.Tech IT, University college of engineering, kakinada, JNTU Kakinada, A.P, India
2. Associative Professor, CSE Department, University college of engineering, kakinada,JNT Kakinada, A.P, India
Satyasri.adapa@gmail.com, krishna.ankala@gmail.com

Abstract: *Nowadays similarity search plays a vital role as a high demanding process in cloud computing and the trend is to outsource such services to 3rd party cloud providers. Outsourcing to a cloud provides advantages such as low initial investments, low storage costs and also provides scalability (more storage or computational power can be added on the fly, when it's needed – so called pay-as-you-go principle).The main aim of this project is collecting the similarity queries from various users and stored in the database for providing security. In this paper a medical related data is collected and saved as the database as headache and diabetes disease related information. Here now all the data's are stored in the hierarchical order in a subject name or age or disease. The cloud computing setting in which similarity querying of metric data is outsourced to a service provider. No one else including the service provider should be able to view the data. Now that data will be kept private. Depending on the queries it will be revealed to the trusted users alone. This transformation technique offers perfect data privacy for the data owner but it gives the final result at multiple rounds of communication. Actually now this technique also provides an interesting trade-off between query cost and accuracy. Already existing solutions either offer query efficiency at no privacy or they offer complete data privacy while sacrificing query efficiency. But the proposed methods are secure and efficient.*

KeyTerms—Query processing, Security, and protection.

1. Introduction:

Advances in digital measurement and engineering technologies enable with more and more data being collected in all kinds of scientific processes (medicine, astronomy, etc.) or commercial applications such as on-line marketing and social networking , searching in large data sets became providing one of the key tasks performed these days. Such data often does not provide efficient meta-data description , so in many applications similarity search is more important than an exact match or keyword search[1]. Since similarity search plays a vital role as a high demanding process in cloud computing and the trend is to outsource such services to 3rd party cloud providers. Outsourcing to a cloud provides advantages such as low initial investments, low storage costs and a good scalability (more storage or computational power can be added on the fly, when it's needed – so called pay-as-you-go principle).

This paper we can see two possible scenarios of outsourcing similarity search. In the first , consider, user has their similarity search technology and wants to use the hardware of a cloud infrastructure provider. In the second, a similarity search service provider makes the technology available for end users so they can use the engine without an actual knowledge of the technology. We observe an increasing trend of the latter case and we will refer to this as similarity cloud.

In both scenarios, users might not want to shows all their data which might be sensitive (e.g. medicine data) or valuable (e.g. data collected from a scientific research), to a third party provider, which is, in general, untrusted. In these cases, privacy of the data is of high performance. Hence the similarity cloud has to provide mechanisms which allow applying privacy requirements of the end user

Now example of valuable data we illustrate the sensitivity issues with more than scenarios. First, consider space programs such as the NASA Apollo program on the Earth's Moon[2] or the ESA Mars Express[3] that collect scientifically valuable data. The data is known to be private before it is released to the public. For example, the time series data is collected from sensors to study the atmosphere's density. Such the data is usually analyzed by the scientists involved in circumstance up the instruments, prior to being made available to general community. At the first stage, access is restrained to authorized scientists for first analysis, because of the substantial efforts investing in building, deploying instruments, and testing, and in refining the data prior to use. Such that valuable data need protection and security when outsourced, to the ensure that the investments by scientific groups are decently rewarded.

2. Related work and existing model:

This paper focuses on the outsourcing of metric data sets proving security. The main aim is to enforce the user authorization specified by the data owner, even when the service provider can be untrusted. It presents techniques that protect location data from attackers, while allowed authorized users to issue queries that are executed efficiently by the service provider. In the literature, a number of concepts for securing databases have been researched. Private information retrieval techniques[1]. In that technique gives query efficiency but it never give the query privacy. Its consider sometimes gives the query privacy while sacrificing the query efficiency and sometimes hide user query to the data owner. If user ask any required query, in that technique just searching the query but never retrieve the query to the responding user. So now there is no guarantee for retrieval and accuracy. It cannot prevent an attacker illegally copying the data from the data set.

Generally, cloud computing providers[4] try to solve any problem by depending on the solutions only provided to the data owner and are not to release outsourced data to untrusted users and 3rd parties[1] . Even those if the provider anyone, without permission the data owner the data is not guaranteed to be safe. Automatically leaks of data are informed regularly, and hackers may still maintains the to gain access to data. So to believe that data owners will be find it attractive to outsource encrypted kind of plain data. This paper introduces existing work on Indexing and nearest neighbour Search in Metric Space data. This technique is working on privacy and security of outsourced data. The main field work privacy-preserving data mining, various techniques which posses few malfunctioning have been developed for easy introducing noise into data, prior to sending the service provider. However, such things does not guarantee the exact retrieval of results. It seems the outsourcing database service to a service provider[4] was introduced by Hacigumus. Since then, various techniques have been developed easy to maintain the confidentiality of outsourced data.

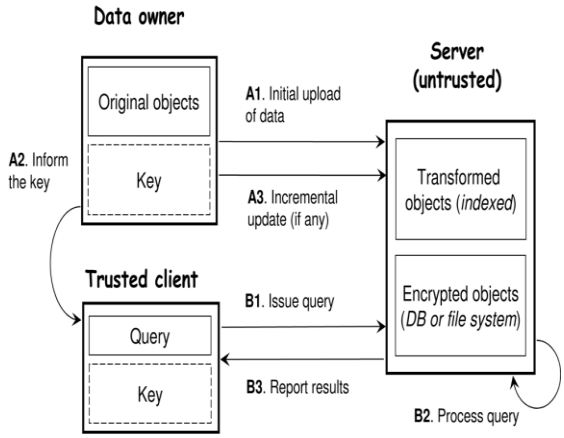


Fig. 1. Scenario overview

It consists of three entities: a data owner, a data owner, a trusted query user, and an untrusted server. On the one is data owner wishes to upload his data to the server so that users are freely execute queries on those data. On the second hand, the data owner trusts only the users and nobody else (including the server).The data owner has a set P of (original) objects and key to be using for transformation. First, start the data owner applies a transformation function (with a key) to convert P into a set P0 of transformation objects, and

uploads the set P0 to the sever (see A1 in the fig).The main server builds an index structure on the set P0 in order to provide facilitate efficient search. In addition, the data owner applies a encryption method e.g., AES on the set of original table (or in the resulting encrypted objects (with their IDs) are uploaded to the server and stored in relational table (or in the file system).second thing the data owner informs every user of the transformation key (see step A2). In the future, enhance the data owner is allowed to perform incremental insertion/deletion of objects (see step A3). At every query time, a trusted user applies the transformation function with a key to the query and then sending the transformed query to the server (see step B1). After then, the server processes the query (see step B2), and reports the results all are back to the user (see step B3).Finally, the user decodes the retrieved results back into the actual results. Observe that these results are contain only the IDs of the actual objects. The user may optionally request to the server to return the actual objects that correspond to the above result set.

To uses the term object for the metric data of interest to the data owner only. A transformed object then the refers to an object obtained from a transformation. Actually this brute-force solution is the one we have mentioned in the Introduction. In this is the offline construction phase, the brute-force is taken multiple rounds working ,the data owner applies the conventional encryption only (e.g., AES) on each and every object and then uploads those encrypted objects to the server.
At query time, the every user needs to download all encrypted objects from the server, decrypt them and then finally the actual result .as mentioned, it is high secure, but its query and communication cost are both prohibitively high, basically this two rounds of communication process and pay-as-you-go is not supported. Actually this anonymization-based solution achieves data privacy by means of K-anonymity the total objects are generalized object cannot be distinguished form K-1 other generalized objects. In the such a way every generalized object cannot be distinguished from k-1 other generalized objects. In the context of Proximity search, it is able to confuse the ranking of transformed objects because K-1 of them have the equal rank as the transformed object of the actual nearest neighbour. Thus, we will still consider this solution as a competitor, even the though k- anonymity is not a suitable privacy and security guarantee for our applications.

The existing solutions also offer query efficiency at no privacy or they offer complete data privacy while giving query efficiency. But now the proposed methods gives a privacy and security. It's high secure and efficient. With this technique having so much of disadvantages to overcome this problem we go for proposal.

3.Encrypted Hierarchical Index Search (EHI):

Encrypted hierarchical index search , this module is always allowed a privacy with query efficiency and query is guarantee to be accurate. Actually first stored in my database only have medical related data such as headache, fever, and diabetes disease related information. Here now all the data's are stored in the hierarchical order in a age wise or subject wise or disease wise. For we taken example two or more people affected by fever so they are asking fever related queries only .
Then the data owner goes for proximity searching operation with need to take help of the EHI algorithm. In First one is searching for fever related queries is accessible in the database or not. Here indexing is vital role it is mainly used for efficient searching of data in the database. Now finally the searching operation is finished successfully. After then goes for fetching operation. The our query fever related available in the database so the data is fetched and finally retrieve the data to the trusted users alone[5].

This method presents a client algorithm, It seems called encrypted hierarchical index, for working NN search on an encrypted hierarchical index stored at the server. This method offers high data privacy for the data owner, but it gives multiple communication round trips during Query processing[1][9]. The transformation key usually send the secret key after client receives secret key is verifying the client . The transformation key of EHI is normal an encryption key CK for encryption algorithm (e.g., AES).

The main query processing technique is used to classify the query and secret key send the trusted client. It is simple to retrieval of information from a data followed the set of retrieval criteria and the data itself remaining not modified .Since the three index stored at the server is encrypted, actually the server cannot process the NN query by itself. An algorithm for communication between the client and the server need to develop in the followed the query correctly. The total responsible time of the algorithm consider the round trip latency and data transfer time. Round trip latency is also called the round trip time. Its not to complete the transfer but so much of time taken send the query to travel from source to destination and acknowledgement. Data transfer time is depending on the size of the data time and transfer time in hard disks. Normally the perfect-first NN search algorithm guarantee that the seek transfer time is minimized. Finally , in the above consider, they need to send a message to the server each time a query is requested. This would incur very high portable round trip latency .

4.Metric preserving transformation:

The above same EHI type of operation is also done here (searching, indexing ,fetching, retrieving). Metric preserving transformation, for assessing the NN[7] query, after that MPT gives the final result at two rounds of communication during the query phase. Here we use only distance bounding phase and candidate retrieval phase by uses that two phases it comes the final result at single rounds of communication. The distance bounding phase main focus is to filter the keyword in the database list. The candidate retrieval phase is also filter the number in the database list. Now here both are using the optimization method is mainly used for reducing the processing overhead and increasing the efficiency. How to work reduce the processing overhead, first is largest database split up into smallest database after merge the database we get the result in single rounds.

5. Flexible distance-based dynamic hashing:

Here we using a new hashing-based technique is called FLEXIBLE DISTANCE-BASED DYNAMIC HASHING, for working the NN query. Actually main focus of this technique is that the server always returns a particular sized candidate set. The candidate set candidate set (in one communication round). The client then refines the candidate set to obtain the total result and also though FDH is not guaranteed to return the exact result, the final result is very near to the actual NN in practice.

On query processing, FDH allows the client to specify an integer parameter Θ for increasing the accuracy of a query result, without including rebuilding the transformed data stored at the server. In our technique FDH method employs a novel technique for conceptually linking similar hash buckets [6], in order process to maximize the utility of the transformed data for answering every queries.The transformation key consists of an encryption Key CK, an integer A, and A pairs of the form (a_i,r_i) where a_i is an object and r_i is a distance value.

The query processing strategy is to apply a proximity search on the above metric space index. The pseudocode of the searching algorithm for FDH. The client using the specifies an additional integer parameter and requests the server to retrieve the tuples whose bitmaps are the closest to the query bitmap BM. After getting the result tuples from the server, the client encrypt them into original objects and computes their distances from q.
The main client refines the candidate set to obtain the final result. This parameter provides a trade-off between the query cost and accuracy . It always gives flexibility to the user. The FDH method gives a final result at single rounds of communication but its only not exact results very close to client result it does not secure the data.

6.Enforcing δ -Gap in Original Space:

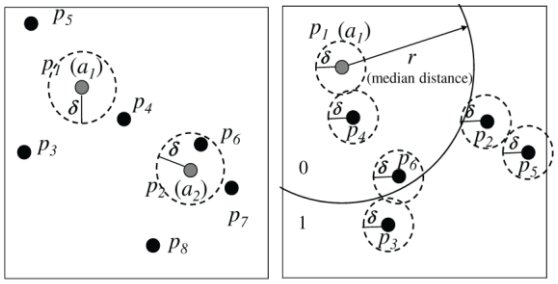
Now the above MPT and FDH is it's don't shows the exact result its very close to client requirements. We now discuss how the δ -gap guarantee since the data owner is able to tune the value of δ such that it describes exactly the required degree of obfuscation ,can be achieved by adapting our transformation methods MPT and FDH. Recall once previous that this guarantee is used to define privacy in the original space, and it requires the actual

result that: each original object metric proximity data must be represented by a tuple such that a is a reference object. Even those requirement has to be fulfilled in the original metric space before applying our transformation method. Similarly we call these extended solutions $\delta - MPT$ and $\delta - FDH$, respectively. Only their algorithms construction are modified; actually the query algorithms of MPT and FDH can be directly reused for processing queries as the schema used for the transformed data remains unchanged.

In order to provide the $\delta - gap$ guarantee in MPT, we modify MPT method as follows ,by restricting how a data object can be assigned to anchor object of the set. Specifically ,we define the set metric space for object. Every anchor in query always satisfies the $\delta - gap$ guarantee with each object. Then we assign object to an anchor form existing object, by using a heuristic (e.g.,finding nearest anchor).

Suppose we now use the above variant to assign objects to anchors in the example fig:2a, where object p1 and p2 center are anchor anchor a_i and its radius as δ. ,To meet the $\delta - gap$ be assigned to the anchor of the center of that circle. For instance ,object and $p_6$are assigned to anchor a_2.Similarly, given figure.2 object p_2 and p_6 are assigned to anchor a_2 and similarly object p_2 and p_6 are assigned to anchor a_2.Every remaining object can be simply assigned to its nearest anchor[8].

We observe that a small δ each value only forces a small number of object to be assigned to requirement ,an object within located cannot be assign to father anchors. To offer the $\delta - gap$ guarantee in FDH,we displace each original objects p, by the distance δ.Here p convert a into bitmap in the transformed space.Fig.2b shows how this variant work. In this example, location within the solid circle are mapped to the bit value0 and ,where any other location are mapped to bit value 1



(a) δ-MPT (b) δ-FDH

Fig. 2 $\delta - gap$ variants of transformation methods.

Every dotted circle represents the possible location of an objects p_i after the displacement. We observe that objects p_1 and p_4 are mapped to the bit value 0,basically regardless of how they are displaced. Similarly ,p_3 and p_5 are always mapped to the bit value 1.Those remaining object(e.g.,p_2,p_6) can be mapped to bit value 0 or 1. Every bit value depending on its displaced location. So here the client request dose not confused the data send the required data. It always gives flexibility to the user. The $\delta - gap$, $\delta - MPT$ and $\delta - FDH$ methods gives a final result at single rounds of communication its exact results very close to client result and secure the data.

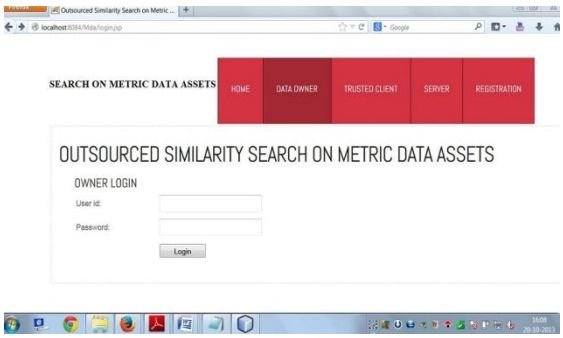


Fig. 3 Data owner login page

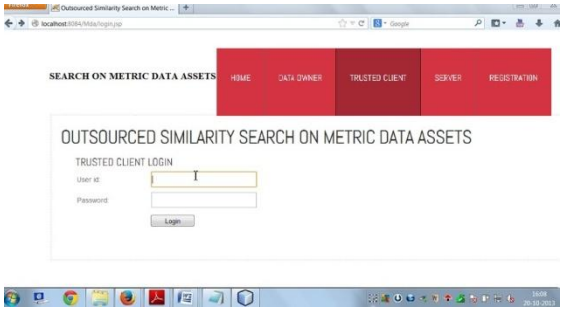


Fig. 4. Client login page

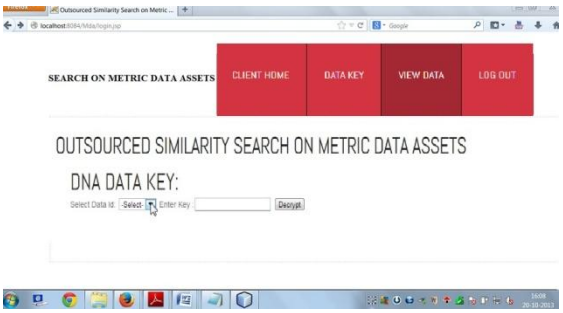


Fig. 5. Data view page

7.Conclusions:

We consider this paper already existing solutions either offer query efficiency at no privacy or they offer complete data privacy while sacrificing query efficiency. But the proposed methods are secure and efficient. The paper presents methods to encode a dataset such that only authorized users can access the content, while the service provider "blindly" evaluates queries , without seeing the actual data. It is important for the data owner to choose an appropriate transformation method that best matches the requirements. The cloud computing setting in which similarity querying of metric data is outsourced to a service provider. No one else including the service provider should be able to view the data It is attractive to be able to maintain data confidentiality with respect to untrusted parties, including the service provider. We are proposing four transformation methods. The introduce first method is encrypted hierarchical index search algorithms gives the final result multiple rounds of communication. We should taken the second method is Metric Preserving Transformation method guarantees correctness of the final search result. But at the taken two rounds of communication. We should taken The third proposed method is Flexible Distance based Hashing methods completed in just a single round of communication but it's taken some confusion to transformation required result. The fourth method $\delta - gap$ is actual result is very close to the exact result. Both $\delta - MPT$ and $\delta - FDH$ are extended to satisfy the $\delta - gap$ privacy guarantee.

8.Acknowledgments:

The paper was implemented during my M.Tech in University College of Engineering Kakinada,under the guidance of A.Krishna Mohan sir As a Associate Professor of department of CSE in JNTU Kakinada.I implemented this paper by using following reference of work shown as below.

References:

[1] M.L. Yiu, I. Assent, C.S. Jensen, and P. Kalnis, "Outsourced Similarity Search on Metric Data Assets," DB Technical Report TR-28, Aalborg Univ., 2010.
[2]http://ti.arc.nasa.gov/project/planetary/mon.
[3]http://sci.esa.int/marsexpress/.
[4] G.R. Hjaltason and H. Samet, "Index-Driven Similarity Search in Metric Spaces," ACM Trans. Database Systems, vol. 28, no. 4, pp. 517-580, 2003.
[5] R. Weber, H.-J. Schek, and S. Blott. A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces. In VLDB, 1998.
[6]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.
[7]Brin, S.: Near neighbor search in large metric spaces. In VLDB pp. 574–584 (1995).
[8] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, "Order-Preserving Encryption for Numeric Data," in
SIGMOD.

[9]T. Bozkaya and Z.M. O ¨ zsoyoglu, "Indexing Large Metric Spaces for Similarity Search Queries,"ACM Trans. Database Systems, vol. 24, no. 3, pp. 361-404, 1999.