Efficient Similarity Search over Encrypted Data

Mohit Kulkarni[1], Nikhil Kumar[2], Santosh Vaidande[3], B.S.Satpute[4]

[1]Mohit Kulkarni: Student, Dept. of Computer Engineering, Dr. D. Y. Patil Institute of Technology, Maharashtra, India.
[2]Nikhil Kumar: Student, Dept. of Computer Engineering, Dr. D. Y. Patil Institute of Technology, Maharashtra, India.
[3]Santosh Vaidande: Student, Dept. of Computer Engineering, Dr. D. Y. Patil Institute of Technology, Maharashtra, India.
[4]B.S. Satpute: Professor, Dept. of Computer Engineering, Dr. D. Y. Patil Institute of Technology, Maharashtra, India.

---***---

Abstract – *In the present time, due to attractive features of cloud computing, the massive amount of data has been stored in the cloud. Though cloud-based services offer many benefits but privacy and security of the sensitive data is a big issue. These issues are resolved by storing sensitive data in encrypted form. Encrypted storage protects the data against unauthorized access, but it weakens some basic and important functionality like search operation on the data, i.e. searching the required data by the user on the encrypted data requires data to be decrypted first and then search, so this eventually, slows down the process of searching. To achieve this many encryption schemes have been proposed, however, all of the schemes handle exact Query matching but not Similarity matching. While user uploads the file, features are extracted from each document. When the user fires a query, trapdoor of that query is generated and search is performed by finding the correlation among documents stored on cloud and query keyword, using Locality Sensitive Hashing.*

Key Words **-** Locality sensitive hashing, Encrypted data, Similarity search, Cloud computing, Data privacy and Query matching.

1. INTRODUCTION

In present data comprehensive environment, cloud computing is common because it removes the burden of enormous data management in a cost effective way. The sensitive data send to untrusted cloud servers will lead to privacy issues. To lessen the worries, sensitive data must be deployed in the encrypted form which avoids illegal access. Though encryption provides protection, it complicates the fundamental search operation. There are many algorithms which support the operation which is called Searchable Encryption Scheme.

Conventionally, almost all schemes are designed for exact query matching. Query matching is the process of finding the user's desired data from the cloud according to the stated feature. But actually, it is more logical to perform retrieval according to the similarity with the stated feature instead of the existence of it.

Similarity search is a problem that optimizes and finds the point in a given set that is closest to a given point. It is predefined that efficient methods are required to do similarity search over the huge amount of encrypted data. The basic algorithm of our project is the approximate near neighbor search algorithm called Locality Sensitive Hashing (LSH). LSH is broadly used for fast similarity search on data in information retrieval. Our project has two parts: User 'Uploading the data' and 'Searching the query'.

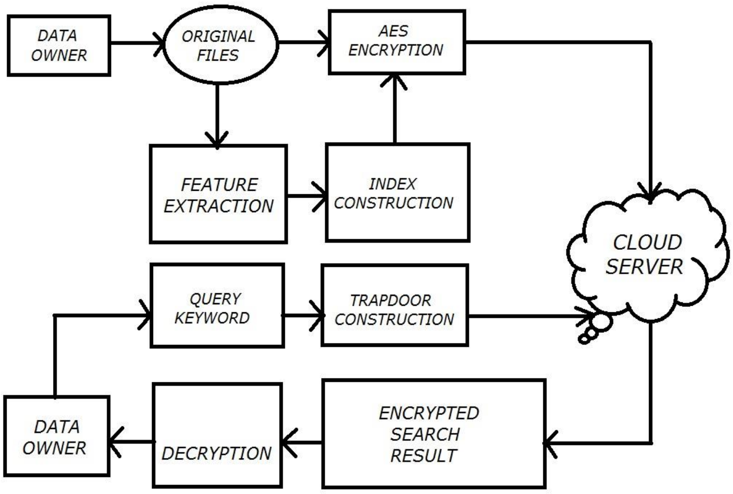

Fig 1: Complete Process

Firstly, the user uploads the data. Then file features are extracted by preprocessing it. Stop words removal and Stemming id is the next process. The words are split at the certain character to form buckets and continue till the total length of the word after which they are stored in a file. After that Encryption of both Plaintext and the Indexed file is done and stored in the cloud.

The second part is sending the file from cloud storage to the receiver. For this, the receiver will send a request for a list of files**.** When the user fires a query, the trapdoor is constructed i.e. **-** it is preprocessed and the bucket of that query is formed. Trapdoor helps to search on encrypted data. Cloud performs a search by using LSH. Document having max similarity to the query will be returned to the user. Then the user can choose the file he wants to download.

Eventually, this proposed system will reduce the time required to get desired data by directly searching over encrypted data.

2. LITERATURE SURVEY

Cong Wang et al. [1] introduces secured rank keyword search for data stored in the cloud. Firstly, they ranked the keywords using effective ranking schemas while maintaining security over the cloud and efficiency of data searched. The ranking is done based on index construction and uses ranking function to give a score to keywords. The major issue in this is computational overhead for ranking keywords.

Cheng and Mitzenmacher [2] offer to search on encrypted data stored remotely using keywords that preserve the privacy of data. Here they describe retrieving the encrypted data using the keyword index (i.e. dictionary) which is created by the user itself. The dictionary is present on the remote server including the file, using which the user can retrieve the encrypted data while maintaining privacy. It requires extra storage to store keyword index.

Jan Li et al. [3] developed schemes to retrieve the encrypted data on the cloud using fuzzy keyword search while preserving the privacy and efficiency. They have developed two schemes (wildcard-based technique and gram-based technique) to construct fuzzy keyword set which produces matching files or closed matching files. The owner stores fuzzy keywords set along with the file which is converted into index form. The efficiency of proposed system can still be further improved for attaining better possible results.

Qin Lv et al. [4] elaborates on multi-probe LHS schema for indexing which helps in efficient similarity matching for searched query and produces best possible results. It uses KNN-algorithm for matching files from multiple buckets for a given input query by the user. Thus it requires few hash tables due to probing multiple buckets and saves storage space required to store a large number of hash tables. They even compared entropy based LSH and multi-probe LSH showing the advantages of multi-probe LSH over entropy based LSH.

Dan Boneh and Brent Waters [5] offers the public-key system that supports comparison queries on encrypted data as well as more general queries such as subset queries. These systems support conjunctive queries which are arbitrary in nature without leaking information on individual conjuncts. In addition, a general framework for constructing and analyzing public-key systems supporting queries on encrypted data. In public key systems, a secret key can produce tokens for testing any supported query predicate. The token lets anyone test the predicate on a given cipher text without learning any other information about the plaintext. It represents the general framework to analyze the security of searching on encrypted data systems.

Mehmet Kuzu [6] proposes an approach which uses Locality Sensitive Hashing (LSH) which is the nearest neighbor algorithm for the index creation. Similar features are put into the same bucket with a high probability due to the property of LSH while the features which are not similar are kept in the different buckets. If the data set is small, then the communication cost and search time required by this scheme is better. But if the database size increases, then the time required for communication and for searching also increases rapidly. They use bloom filter for translation of strings. But the disadvantage of this structure is that it is a probabilistic data structure.

3. CONCLUSION

Thus we can conclude that to get an efficient similarity searchable encryption scheme we need to use locality sensitive hashing for fast similarity search in high-dimensional spaces for plain data. In such a context, it is also very critical not to sacrifice the confidentiality of the sensitive data while providing functionality. Use LSH based secure index and a search scheme to enable fast similarity search in the context of encrypted data. Finally, show the performance of the proposed scheme with empirical analysis on a real data.

REFERENCES

[1] Cong Wang, Ning Cao, Jin Li, Kui Ren and Wenjing Lou, “Secure Ranked Keyword Search over Encrypted Cloud Data”, IEEE 30th International Conference on Distributed Computing Systems, 2010, pp. 253-262, doi:10.1109/ICDCS.2010.34.

[2] Yan-Cheng Chang and Michael Mitzenmacher, “Privacy Preserving Keyword Searches on Remote Encrypted Data”, in Proc. of ACNS’05, 2005, pp. 442-455, Springer Berlin Heidelberg.

[3] Jan Li, Q. Wang, C. Wang, N. Cao, K. Ren, W. Lou, “Enabling Efficient Fuzzy Keyword Search over Encrypted Data in Cloud Computing”, Proc. of IEEE INFOCOM’10 Mini-Conference, March 2010, pp. 1-5, IEEE.

[4] Qin Lv, William Josephson, Zhe Wang, Moses Charikar, Kai Li, “Multi-Probe LSH: Efficient Indexing for High Dimensional Similarity Search”, in Proceedings of the 33rd international conference on very large databases, September 2007, pp. 950-961, VLDB Endowment.

[5] Dan Boneh and Brent Waters, “Conjunctive, Subset and Range Queries on Encrypted Data”, in Theory of Cryptography Conference, February 2007, pp. 535-554, Springer Berlin Heidelberg.

[6] Kuzu Mehmet, Mohammad Saiful Islam, Murat Kantarcioglu, “Efficient Similarity Search Over Encrypted Data”, in Data Engineering (ICDE), 2012 IEEE28th International Conference, IEEE, 2012.