International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 04 Issue: 01 | Jan -2017 www.irjet.net p-ISSN: 2395-0072

Privacy Preserving Multi-keyword Top-K Search based on Cosine Similarity Clustering

Sushant Y. Kamble , R. P. Mirajkar

[1]*Computer Science and Engineering Dept., Bharati Vidyapeeth's College of Engineering, Kolhapur (District), Maharashtra-416012, INDIA sushantykamble@gmail.com*
[2]*Asst.Prof. at Computer Science and Engineering Dept., Bharati Vidyapeeth's College of Engineering, Kolhapur (District), Maharashtra-416012, INDIA rahulmirajkar982@gmail.com*

Abstract - *Cloud computing provides the facility to store and manage data remotely. The volume of information is increasing per day. The owners choose to store the sensitive data on the cloud storage. To protect the data from unauthorized accesses, the data must be uploaded in encrypted form. Due to large amount of information is stored on the cloud storage; the association between the documents is hiding when the documents are encrypted. It is necessary to design a search technique which gives the results on the basis of the similarity values of the encrypted documents. In this paper a cosine similarity clustering method is proposed to make the clusters of similar documents based on the cosine values of the document vectors. We also proposed a MRSE-CSI model used to search the documents which are in encrypted form. The proposed search technique only finds the cluster of documents with the highest similarity value instead of searching on the whole dataset. Processing the dataset on two algorithms shows that the time needed to form the clusters in the proposed method is less. When the documents in the dataset increases, the time needed to form clusters also increases. The result of the search shows that increasing the documents also increases the search time of the proposed method.*

Keywords: Cloud computing, multi-Keyword search, cosine similarity clustering, encrypted data

1. INTRODUCTION

Cloud computing becomes popular as it provides huge storage space and high quality services. The large amount of data is created per day. It is a difficult task for the owner of the data to store and manage this large amount of data. To overcome this difficulty, the data owners can store their data on the cloud server to use the on demand applications and services from shared resources [1]. The cloud server providers agreed that their cloud service is armed with strong security constraints though security and privacy are major hindrances which avoid the use of cloud computing services [2]. To protect the sensitive data on the cloud server from unauthorized users, the data owners may encrypt the documents and uploads to cloud server [3]. In the earlier various strong cryptography methods were used to design the search techniques on the cipher text [4], [5], [6]. These techniques needs many operations and require large amount of time. So these techniques are not suitable for big data where information volume is huge. The property of a document depends on its association with other documents. Therefore maintaining the relationship between documents is important to fully express a document.

The results of search returned to the users may contain damaged information due to hardware failure or storage corruption. Thus a mechanism should be given to users to check the accuracy of the search results.

The proposed architecture of search technique is based on the cosine similarity clustering which maintain the association between plain text and encrypted text to improve the efficiency of search.

2. LITERATURE REVIEW

Chi Chen and Xiaojie Zhu [7] used a hierarchical clustering method to maintain the close relationship between plain documents and encrypted documents to increase search efficiency within a big data environment. They also used a coordinate matching technique [8] to measure the relevance score between query and document. They did a model for the efficient multi-keyword ranked search and maintain the privacy of documents, rank security and relevance between retrieved documents.

Jiadi Yu and Peng Lu [9] focused on the problems of the cipher text search using Searchable Symmetric Encryption (SSE) [10], [11]. This SSE technique helps data users to retrieve the documents over the encrypted documents. In Two Round Searchable Encryption (TRSE), they used the similarity relevance concept to solve the privacy issues in searchable encryption. They also showed server side ranking according to order preserving encryption (OPE).

N. Cao, C. Wang and M. Li [12] used "inner product similarity" concept which can find the similarity measure of the information and the keywords of search.

Ruksana Akter, Yoojin Chung [13] defined an evolutionary approach based on cosine similarity clustering. A document vector is used to create the index of every document. The cosine values between the document vectors are calculated. Clusters of the most relevant documents are formed on the basis of the cosine values. Another good feature of their work is that they do not require

specifying the all clusters to be made in advance as most of the available methods.

3. DESIGN ASPECTS

3.1 Proposed System

We used the cosine similarity clustering algorithm to check the time required for the search results. We have developed a search technique which is based on cosine similarity clustering algorithm to improve the efficiency of search.

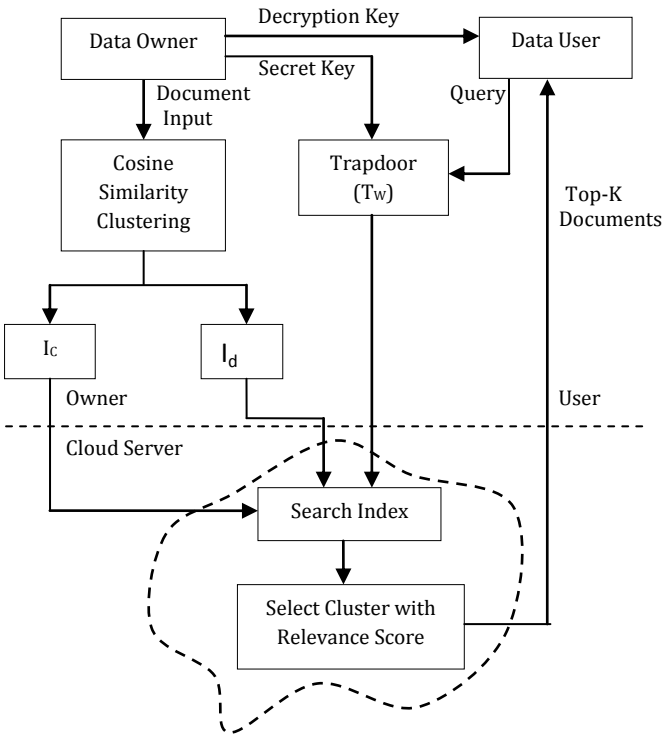

Fig - 1: Architecture of Proposed System

3.2 Methodology

Step 1: **Key Generation: -** The owner of the data generates a key using the size of dictionary and a random generator. This secret key is used for encryption of the document index.

Step 2: Encrypted Index Generation: - The document vector is prepared and applies the cosine similarity clustering algorithm to form the clusters of relevant documents. Every cluster has its cluster index. These cluster indices are encrypted using matrix multiplication with the secret key. These encrypted cluster indices are uploaded to the cloud server.

Step 3: Document Upload: - The documents are encrypted using AES algorithm. These encrypted documents are uploaded to the cloud server.

Step 4: Trapdoor Generation: - The data user send query to the data owner. After analyzing the keywords of query, data owner builds query vector. This query vector is encrypted using matrix multiplication with secret key and send to the user.

Step 5: Document Searching: - The user uploads this encrypted query vector (trapdoor T_W) to the cloud server. The similarity value between trapdoor and encrypted cluster indices (I_C) is calculated by the cloud server. The most relevant cluster with the highest similarity value is selected. This cluster is extracted and again the similarity value is calculated between trapdoor and each document vector. The top-k documents will be returned to the user on the basis of similarity value.

Step 6: Decrypt Documents: - The search result contains the documents in encrypted form. The data user must send request for the key to the owner of data. Data owner sends the key to decrypt the documents.

4. SNAPSHOTS

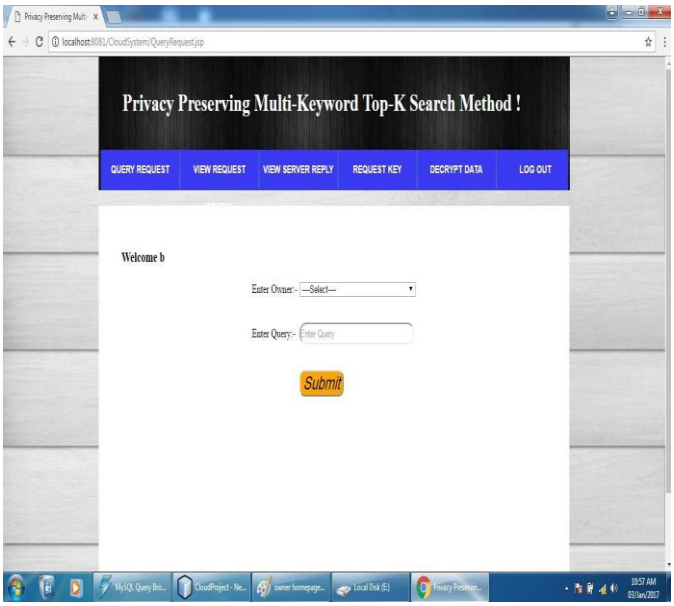

Fig -2: Snapshot of Query Request Page

Fig. 2 shows the query request page. The data user may enter the name of the data owner and keywords of the query and send to data owner.

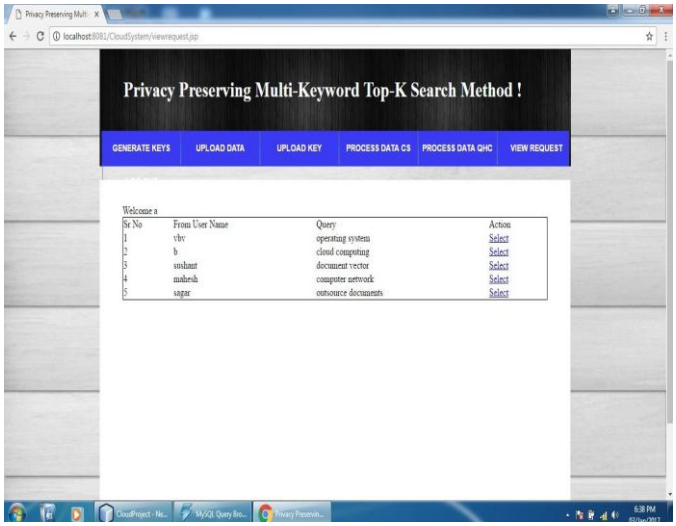

Fig -3: Trapdoor Generation Page

Fig. 3 shows the requests from the data user. Data owner generates the trapdoor and send back to data user.

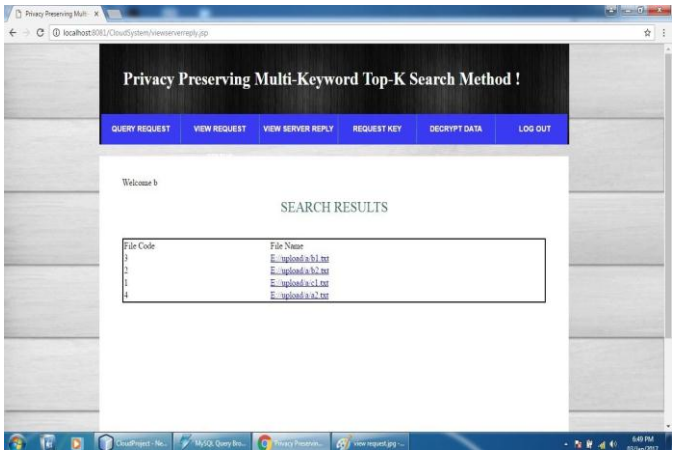

Fig - 4: Search Results Page

Fig. 4 shows the results of search returned from cloud server. These are the most relevant top-k documents to the search keywords.

5. RESULT AND DISSCUTION ON MODULES

Table-1: Search time using two algorithms

Dataset	CS Time (ms)	QHC Time(Ms)
5	51	52
10	57	59
15	61	70
20	66	81
25	73	102

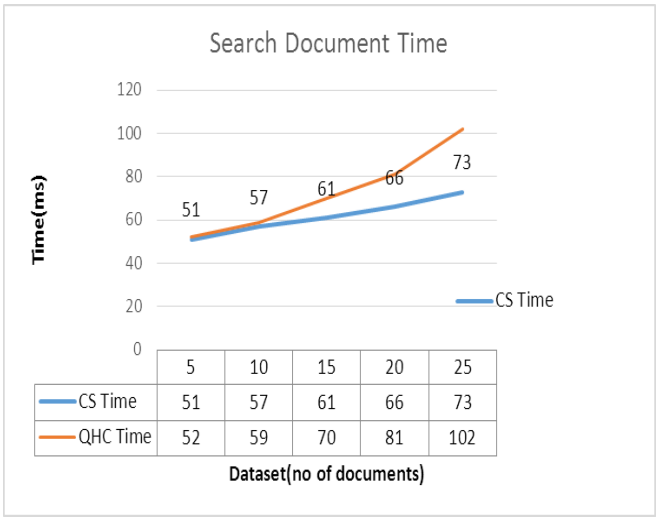

Chart -1: Search Document Time

In the first experiment, document search time is calculated. Five different number of dataset sizes are chosen in the experiment to show the effect on the efficiency of the search results. From chart 1, we can see that the time needed to search the documents increases when the size of dataset increases. Compared with the previous related work [7] time needed to search the documents is less.

Table -2: Time cost to form clusters

Dataset	QHC Time (ms)	CS Time (ms)
5	1934	1731
10	2132	1891
15	2250	1931
20	2472	2112
25	2790	2312

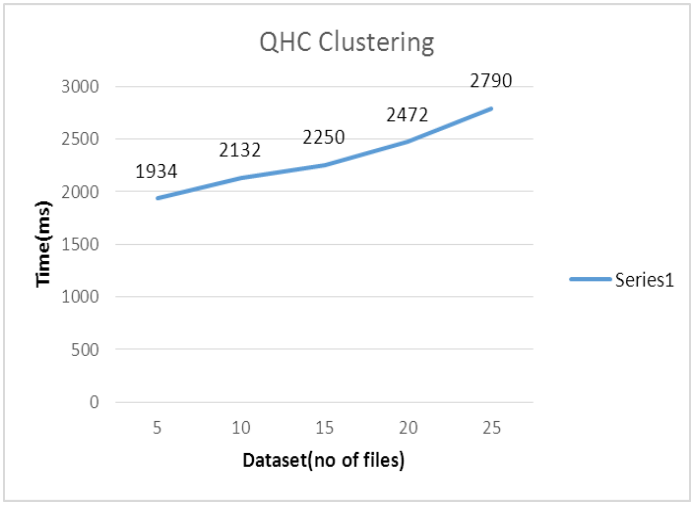

Chart - 2: Time required for Hierarchical clustering

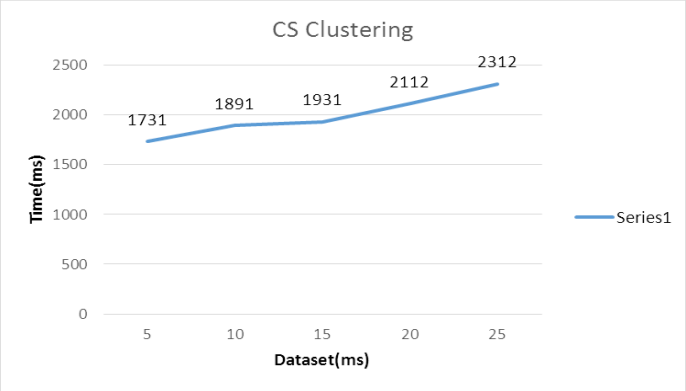

Chart - 3: Time required for Cosine Similarity Clustering

We have developed a system which is based on two different clustering algorithms (i.e. Hierarchical Clustering and Cosine Similarity Clustering)

In the second experiment, we have calculated the time required to form the clusters of relevant documents using hierarchical clustering and Cosine Similarity Clustering algorithms. Five different numbers of dataset sizes are chosen in the experiment to show the effect on the efficiency of the clustering algorithms.

From the charts 2 and 3, we can say that time needed to form clusters of relevant documents in Cosine Similarity Clustering is less as compared with Hierarchical Clustering.

6. CONCLUSIONS

We have developed a MRSE-CSI model based on cosine similarity based clustering and word relevance technique. We analyze the search efficiency of the system using two clustering algorithms. The experimental result proves that the time needed to form the clusters of relevant documents is reduced by using cosine similarity clustering. The experimental result also proves that the speed of the search increases by using the cosine similarity clustering algorithm. The proposed architecture improves the search efficiency and rank security.

REFERENCES

[1] L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, "A break in the clouds: towards a cloud definition," ACM SIGCOMM Comput. Commun. Rev., vol. 39. No. 1, pp. 50-55, 2009.

[2] S. Grzonkowski, P. M. Corcoran, and T. Coughlin, "Security analysis of authentication protocols for next-generation mobile and CE cloud services," in Proc. ICCE, Berlin, Germany, 2011, pp. 83-87.

[3] S. Kamara and K. Lauter, " Cryptographic cloud storage," in RLCPS, January 2010, LNCS. Springer, Heidelberg.

[4] G. Craig. "Fully homomorphic encryption using ideal lattices." STOC. Vol. 9. 2009

[5] D. Boneh, G. Di Crescenzo, R. Ostrovsky, et al. Public key encryption with keyword search[C].Advances in Cryptology-Eurocrypt 2004. Springer Berlin Heidelberg, 2004: 506-522.

[6] D. Cash, S. Jarecki, C. Jutla, et al. Highly-scalable searchable symmetric encryption with support for boolean queries [M].Advances in CryptologyCRYPTO 2013. Springer Berlin Heidelberg, 2013: 353-373.

[7] Chi Chen, Xiaojie Zhu, Peisong Shen, J.Hu S.Gue, Z.Tari and Albert Y. Zomaya, "An Efficient Privacy Preserving Ranked Keyword Search Method," IEEE Transactions on Parallel and Distributed Systems, 2015.

[8] I. H. Witten, A. Moffat, and T. C. Bell, Managing gigabytes: Compressing and indexing documents and images, 2nd ed., San Francisco: Morgan Kaufmann, 1999.

[9] Jiadi Yu, Peng Lu, Yanmin Zhu, Guangtao Xue and Minglu Li, "Toward Secure Multi-Keyword Top-K Retrieval over Encrypted Cloud data, "IEEE Transactions on Dependable and secure computing, Vol. 10, No. 4 July/August 2013.

[10] D. Song, D. Wagner, and A. Perrig, "Practical Techniques for Searches on Encrypted Data," Proc. IEEE Symp. Security and Privacy, 2000.

[11] D. Boneh, G. Crescenzo, R. Ostrovsky, G. Persiano, "Public-Key Encryption with Keyword Search", Proc. Int'l Conf. Theory and Applications of Cryptographic Techniques (Eurocrypt), *2004.*

[12] N. Cao, C. Wang, M. Li, K. Ren, and W. J. Lou, "Privacy-Preserving Multi-keyword Ranked Search over Encrypted Cloud Data," in Proc. IEEE INFOCOM, Shanghai, China, 2011, pp. 829-837.

[13] Ruksana Akter, Yoojin Chung, "An Evolutionary Approach for Document Clustering," 2013 International Conference on Electronic Engineering and Computer Science.