International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Impact Factor (2012): 3.358

Improved Indexing and Advanced Relevance Ranking Score for Multi-Keyword Search over Encrypted Cloud Data

Amol D. Sawant[1] , Prof. M.D Ingle[2]

[1]PG Student, Department of Computer Engineering, Jaywantrao Sawant College of Engineering,
Hadapsar , Pune-411028, India

[2]Associate Professor, Department of Computer Engineering, Jaywantrao Sawant College of Engineering,
Hadapsar, Pune-411028, India

Abstract: ***Cloud computing is used to outsource the large volume of and important o most sensitive data on the remote server that is cloud server. To provide the data confidentiality and privacy, the sensitive cloud data have to be outsourced in encrypted format on commercial public cloud or private cloud. Traditional encryption techniques is used to securely search over encrypted data through single keyword search with rank score of the files also it supports the multiple keyword search but the sum of relevance score of the files is preserved and will not meet the effective data utilization need and the requirement of the large number of users and the large database. In this paper we solve the problem of preserving sum of multiple keywords search files ranking score. The existing ranking algorithm doesn't use the OR of multi-keywords. Also we enhance the relevance ranking score of the files which enhances the system usability by giving relevance ranking instead of sending undifferentiated results. We explore the statistical measure approach i.e. improved relevance score , from the information retrieval to build the index of files and develop the advanced ranking function and the Advanced search to protect the sum of the sensitive score information by improving the index structure. The resulting design able to do the multiple keyword searches without losing the privacy of the multiple keywords and prevent the sum of the relevance score of multi-keyword from the server from leakage. Through analysis this solution gives the strong security guarantee for the compared to the previous searchable encryption scheme for multi-keyword search using OPM function. The experimental result demonstrates the efficiency of the proposed solution, with it reduces the number of distinct keywords and results in reduction of index size and improves the relevance ranking score function and gives the most relevant document to users.***

Keywords: indexing, multi-keywords, relevance ranking, encrypted searching.

1. Introduction

We are using the highly networked environment to store the huge amounts of data in remote place where security of the data is the most important issue, the remote servers and database holders may not be the trusted servers or databases. There are many privacy issues related to access the data from such servers two of them are: sensitivity of i) keywords used in the search queries and ii) the data relevant data retrieved by the user both need to be hidden. As the data is very sensitive and the cloud servers are not the trusted one, they may leak the information to the unauthorized entity. so the data search keywords need to be secured from the servers.

Today we need high storage and computation power which tends to outsource their private and sensitive data and services to clouds. There three types of cloud in access nature Private cloud, Public cloud and Hybrid cloud. Cloud enables the customers to remotely store and access their private data by minimizing the cost of hardware ownership while providing robust and fast services. To provide the privacy to the cloud data the data need to be encrypted and stored on the cloud servers. The data utilization is the very challenging task when the data is encrypted. The data owners may share their outsourced data with the authorized users who are interested to retrieve the specific data files. One of the ways to do this is to use keyword based search which allows users to search the files and retrieve it by using the plaintext search. Data encryption restricts to the user to use plaintext keywords. The keywords need to be encrypted and need to use encrypted keyword search. Previous encryption search allows data to be securely searched over cloud data. Our aim is to achieve an efficient cloud system where any authorized user cannot perform a search on a remote database with multiple keywords, without showing to serve the keywords he does not search for nor the contents of the documents he retrieves.

The existing relevance ranking score function gives the wrong result of the ranking of the documents. As existing algorithm doesn't consider the distance between the keywords in a document that is the relation between the keywords so it gives the wrong result of ranked documents. So we improve the relevance ranking function for ORed of Multi-keyword search which gives better relevance ranking score and more relevant documents that are relevant to search query.

This system is able to perform multiple keyword searches in a single query and ranks the results so the user can retrieve only the top matches and the time of users will save in retrieving the unnecessary data. The relevance score of the file need to secure from the server. Wang C. [2] uses the one to many orders preserving mapping to prevent the file relevance score from the server for single keyword search. If we need to use the multiple keyword search then the sum of the relevance score of the files need to be calculated to retrieve the files. The need is to prevent the sum of the relevance score of the files for the multiple keywords. So the

Licensed Under Creative Commons Attribution CC BY

proposed system is able prevent the sum of the relevance score of the multiple keyword files. So to achieve the Secure multiple keyword search by improving the index creation function and to improve the Ranking function which prevents the leakage of the relevance score of files and enhances the relevance score. The proposed system improved the index for multi-keyword search, it also reduces the number of distinct keywords and we use multilevel indexing to save the searching time and the to give the advanced ranking of the document.

2. Literature Survey

The problem of the multi-keyword search is to provide relevance score for file considering the multiple keywords and the searching time is required more. The Gengiz and Savas [1] have produced the TF and IDF for calculating total relevance score is not efficient score ranking function for multi-keywords and is unable to implement the multi-keyword disjunctive Boolean operation. The index also contains the single keyword file score. For every file one new index is required so to search the index need to all the indexes that requires the more time. So need to improve the index structure and the relevance ranking function.

Wang C.[2] uses the one to many order preserving mapping to prevent the file relevance score from the server for single keyword search. If we need to use the multiple keyword search then the sum of the relevance score of the files need to be calculated to retrieve the files. The need is to prevent the sum of the relevance score of the files for the multiple keywords. So we improve the indexing and ranking function to prevent the sum of score of multiple keywords. We improve the system for multi-keyword search in terms of the time to search and the advanced relevance score.

3. Problem Definition

The problem of keyword search on cloud server is that the server may know the search terms and the privacy of the keyword will not be secure. The search should be done in encrypted format and the data should be stored in encrypted format. The encryption keys should not known to the database server.

The problem is to hide the sum of the relevance score of the multiple keywords in multi-keyword search. To do the multi-keyword search the sum of the relevance score of keywords of files is preserved from the server. To do the ranking of the multiple keywords the server need to compute the sum of the relevance score of each keyword and then comparing the sum server will compute the ranking of files to show to users. If the server knows the relevance score then it will identify the pattern and the result is leakage of information to server. So we use the relevance score preserving algorithm for multi-keyword search. The existing relevance ranking function doesn't provide the correct result of ranking. The existing algorithm doesn't use IR techniques to reduce the number of distinct keywords. The existing indexing algorithm doesn't provide the efficient search of index.

There are three roles in this system architecture are as follows:

1. Data owner, who is actual database owner creates the database and is unable to handle the database, so he stores the data in encrypted format on the cloud server.
2. Users, who are the authorized users of the data owner, he uses the services provided by the data owner through cloud server. Users search the data on cloud server using the encrypted keywords and access the data permitted by the data owner.
3. Cloud Server, is a main cloud database server and maintains the large amount of encrypted cloud storage and handles the requests of the number of users and provide the trusted service to the data owner and makes the encrypted searching of the document and the documents are retrieved.

The overview of the proposed system is illustrated in Figure1. Assume that the parties are semi-honest and do not collude with each other to bypass the security measures.

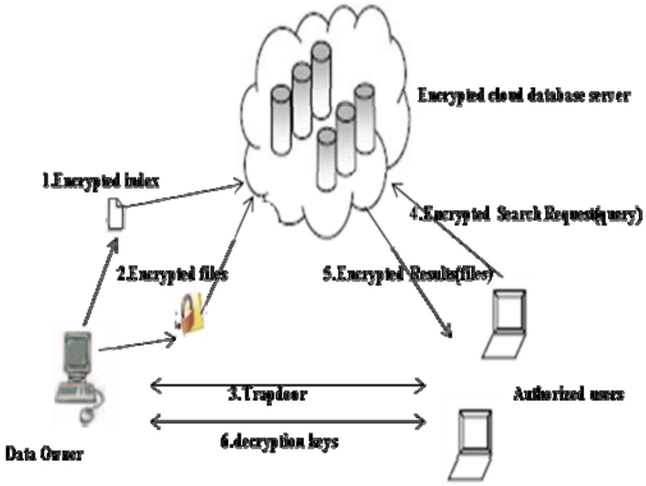

Encrypted cloud database server
1.Encrypted index
4.Encrypted Search Request(query)
2.Encrypted files
5.Encrypted Results(files)
3.Trapdoor
Authorized users
6.decryption keys
Data Owner

Figure 1: Architecture for Encrypted Multi-keyword Search System

In Figure 1, the steps and the interaction between the data owner, User and the cloud Server is illustrated as follows. In the offline stage the data owner creates a search index for each document and one index for all documents. The index is encrypted using the secret keys and the trapdoor is created and is shared between the data owner and the authorised users. The Data owner encrypts the files, then he uploads these search index and the encrypted files to the data server. The use of the encrypted keys provides the security to the documents.

This process is called the index generation and the trapdoor generation as shown in figure1 in step.1 and step 2 when a user wants to perform a keyword search, he first connects to the data owner and requests the trapdoor from the data owner as in step.3 and it takes the trapdoor and generated the encrypted query. As shown in step .4 the user request the data from the cloud server. The cloud servers makes the encrypted search and rank the corresponding documents and send the encrypted result to the user as in step.5 in figure 1. Then the user request the decryption keys from the data owner as shown in the step.6 and get the decryption keys for the files and decrypt the required files.

Licensed Under Creative Commons Attribution CC BY

4. Research Methodology

4.1 Domain Index

The Cong Wang [2] has used the one to much order preserving technique to provide the privacy to the relevance score of single keyword. The previous secure ranked multi-keyword search does not provide the security to the sum of the relevance score. so we use improve the index structure and the proposed index contains the domains for the documents and the levels of the index for correct math and the posting list of files for each correct match and the documents are ordered with encrypted relevance score. Our new contribution is to improve the relevance score criteria of the file with keyword and to improve the result of the ranking to the user and user able to locate the relevance file that he need. The domain and the index creation algorithm is as follows: the improved index structure for the multi-keyword search is as in figure. 2

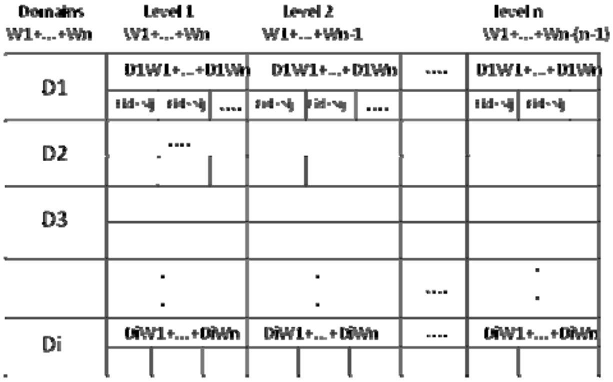


Figure 2: Design of Domain Multi-level Index for Multi-keyword Search

The index structure contains the domains and the each domain contains the multilevel index level 1 to level n. Level n index contains the single keyword index and its file ids with corresponding ordered encrypted relevance scores.

Algorithm to create domain-index

Algorithm domain-Index (K, C)

1) CreateDomain()
a. Scan C with advanced IR techniques such as stemming, stop words and case folding and find the distinct keywords. W=(w1,w2,....,wm)
b. Create the domain using set of distinct keywords and create the list of keywords and files for every domain.
D1={(wd11,.............wd1n),(fd11,......fd1n)}
Di={(wdi1,..............wdin),(fdi1,......fdin)}

2) Encryptdomain()
a. bit conversion of each keywords and Create Index for the each domain using the disjunctive OR operation of the set of keywords. (wdi1 ...wdin) for one domain.

3) CreateMulti-levelIndex()
Create level 1 to level n index for the set of domain keywords (wdi1 ...wdin)
level 1 ...disjunctive OR of ‘n’ keywords
level 2 ...disjunctive OR of ‘n-1’ keywords
.
level n ...disjunctive OR of ‘n-(n-1)’ keywords

4) Build Posting List
For (each level.i=1...n)
For (j=1...n file)
Calculate the total relevance score Sij for multi-keyword function and encrypt the relevance score

5) Secure Index
create(id)(Fi)|| (wli)(Sij)
for all Level index L(Wi) 1<=i<=m
encrypt the all the entries with l' padding
o's <<0l k id(Fij) k OPMfz(Sij)>>

5. Mathematical Model for Relevance Score function

The relevance ranking function given by Cengiz and Savas[1] is as follows which doesn’t consider the distance between keywords. If the distance between the keywords is more than the file is not more relevant for set of multi-keywords. If the distance between the multiple keywords is less than the file is more relevant and so the score of the relevance is more. The old relevance ranking score function is as follows:

$$Score\,(Q,Fd)=\sum\nolimits_{t\epsilon Q}\frac{1}{Fd}(1+In\,Fd,t)In\left(1+\frac{N}{Ft}\right) \text{ ... (1)}$$

The improved new relevance ranking Score improvement for multi-keyword search is as follows, which calculates the sum of the distanced between the keywords, which gives more relevance of the file.

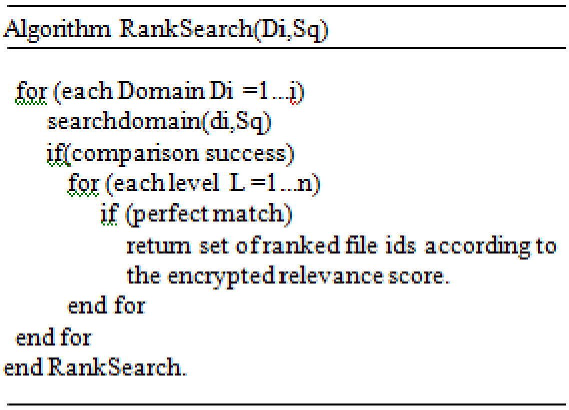
$$Score(Q,Fd)=\sum_{t\epsilon Q}^{m}\{\frac{1}{Fd}(1+In\,Fd,t)In\left(1+\frac{N}{Ft}\right)\}\frac{m}{In(\sum_{t\epsilon Q}^{m}dist(Qt,Qt+1))} \text{ .... (2)}$$

where, Q denotes the search keywords, $f_{d,t}$ denotes the TF of term t in file F_d, N denotes the total number of files. F_d denotes the length of the file by counting number of index terms. f_t denotes the number of files contains the term t, the dist(Q_t, Q_{t+1}) denotes the distance that is the number of keywords between the t and t+1 distinct word and *m* is number of keywords in the query.

The proposed module contains the three entities. The data owner entity having following algorithms to generate the improved index and to generate the improved relevance ranking algorithm.

Licensed Under Creative Commons Attribution CC BY

5.1 Multi-Keyword Searching Algorithm



```
Algorithm RankSearch(Di,Sq)

 for (each Domain Di =1...i)
     searchdomain(di,Sq)
     if(comparison success)
        for (each level L =1...n)
            if (perfect match)
               return set of ranked file ids according to
               the encrypted relevance score.
        end for
 end for
end RankSearch.
```

Where Di is the domain of the keywords, so Sq is the query index submitted by the user, L is the set of levels. The above multi-keyword Rank Search algorithm gives the perfect match for the multiple keywords with ranked documents. The first comparison is made with the domain from domain 1 to i. So i comparisons are required then the depending on the number of keywords in a query the level comparison is made. If there are n numbers of keywords then 1 comparison is made and if one keyword is there then n comparisons are required.

6. Results

Consider there are 1000 files in a collection of cloud, and then the comparison is shown in table 1.given bellow.

Number of File: 10000
Number of Domains: 0
Number of keywords/levels in search query: 3
The number of comparisons required to rank search are 1010.
New index Searching is as follows:
Number of files: 10000
Number of Domains: 10
Number of keywords/levels in search query: 3
The number of comparisons required to rank search are 113.

Table 1: Number of Comparisons for old index search and the proposed index search.

Type of index	*No. of Domains*	*No. of keywords /levels*	*Number of Comparisons*
Old index search	--	-	1010
New Domain index search	10	3	113

The use of improved indexing algorithm reduces the number of distinct keywords from the file, which intern reduces the size of the index and time of the index search.

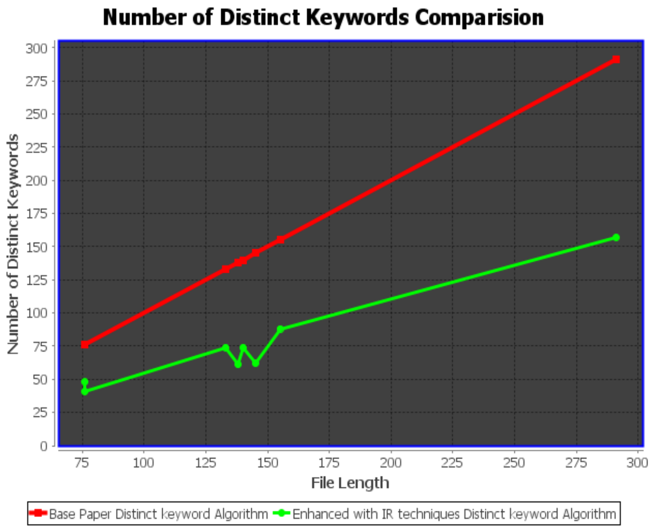


Figure 3: Result analysis of the create index for extracting distinct keywords

As shown in figure.3 the number of distinct keywords will reduces with increase in the file size. As the distance between keywords increases the score decreases that relevance reduces and the difference between keywords decrease the score increases means the more relevance file. In figure.4 shows the result if the improved relevance score calculating function. As in figure if the distance between the number of keywords is 5 the score is 60 and above and if the distance is 25 the relevance score is 10.

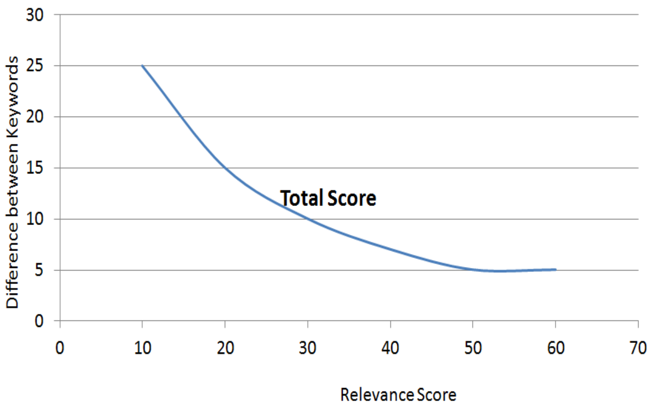


Figure 4: Result Analysis for the calculating Score

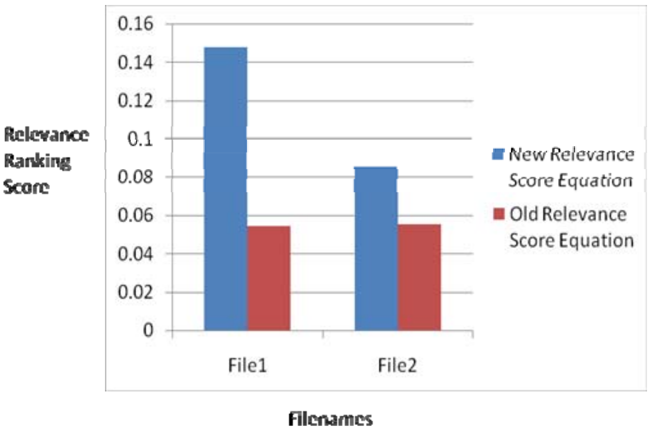


Figure5: Result of Relevance Ranking Score of files for (interpr, firm, organ) multi-keyword

Licensed Under Creative Commons Attribution CC BY

The figure 5 shows the relevance ranking score comparison of files for (interpr,firm,organ) multi-keyword and shows the comparison between New Relevance Ranking Score Equation and Old Relevance Ranking Score Equation. Old Relevance ranking equation gives the wrong result as File2 is having more relevance score than the File1, but File2 is not more relevant than File1, which results in wrong result. New Relevance Ranking equation gives the correct result.

7. Conclusion

The previous ranked search is a single keyword search with relevance score preserving. To produce the multi-keyword search it is necessary to hide the relevance score when ranking is done at server inside. We solve the problem of privacy of the total relevance score of the file. We do the ranking at the data owner in the index itself that results in privacy of sum of the relevance rank score of files. The use of IR techniques reduces the number of distinct keywords which results in less index size. The advanced relevance ranking of multi-keyword search is produces the better ranked result of the documents. The domain index reduces the searching time of the keyword or document.

8. Future Scope

We also improve the relevance ranking function to improve the total score of the file. The ranked search function greatly enhances the relevance of the returned result and it reduces the communication overhead, which is highly desirable. The conjunctive normal search is implemented to implement the multi-keyword search. The test result will produce the reduction in the distinct keywords using advanced IR techniques, also improves the relevance ranking score of the files. As we are using the multi-keyword search, search requires the more time than single keyword search, here future enhancement can be to reduce the search time**.**

References

[1] Cengiz and Savas “Efficient and Secure Ranked Multi-Keyword Search on Encrypted Cloud Data” PAIS 2012, March 30, 2012, Berlin, Germany Copyright 2012 ACM 978-1-4503-1143-4/12/03

[2] Cong Wang, IEEE, Ning Cao, "Enabling Secure and E_cient Ranked Keyword Search over Outsourced Cloud Data" IEEE Transactions on Parallel and Distributed Systems Vol.23 No.8 Year 2012.

[3] Ming Li, Utah State University Shucheng Yu, Hou, Virginia Polytechnic Institute and State University "Toward Privacy- Assured and Searchable Cloud Data Storage Services"0890- 8044/13/25.00 2013 IEEE.

[4] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou “Privacy-preserving multi-keyword ranked search over encrypted cloud data." In IEEE INFOCOM, 2011.

[5] D. Boneh, E. Kushilevitz, R. Ostrovsky, and W. Skeith.”Public key encryption that allows pir Querie "Advances in Cryptology – CRYPTO" 2007, volume 4622 of Lecture Notes in Computer Science, pages 5067. Springer Berlin / Heidelberg, 2007

[6] Y.-C. Chang and M. Mitzenmacher. “Privacy Preserving Keyword Searches on Remote Encrypted Data, In Applied Cryptography and Network Security”, pages 442455. Springer, 2005

[7] Wang, N. Cao, J. Li, K. Ren, and W. Lou, “Secure keyword search over encrypted cloud data.” *www*.acm.org/citation.cfm?id=1949221.1949237.

[8] Singhal, “Modern information Retrieval : A brief overview ” IEEE Data engineering Bulletin , Vol. 24, no.4, pp.35-43,2

[9] S. Zerr, D. Olmedilla, W. Nejdl, and W. Siberski, “Zerber+r:Top-kretrieval from a confidential index,” in Proc. of EDBT’09, 2009

[10] C. Wang, S. Chow, Q. Wang, K. Ren, and W. Lou, “Privacy-Preserving Public Auditing for Secure Cloud Storage,” IEEE Transactions on Computers (TC)

[11] RFC, “Request for Comments Database,” http: //www.ietf.org/rfc.html.

Author Profile

Mr. Amol Sawant received BE CSE Degree in 2008 from Solapur University and he is student of Jayawantrao Sawant College of Engg. Doing ME in Computer and Currently Working in the JSPM’s Jayawantrao Sawant Polytechnic Hadapsar,

Prof. M. D. Ingle, He received MTECH Post Degree in 2005 from Mumbai University and he is currently working as Associate. Professor in JSPM’s Jayawantrao Sawant College of Engg. Hadapsar and Also he is Working as a ME Computer Coordinator in JSCOEngg.

Licensed Under Creative Commons Attribution CC BY