Empower syntactic exploration based on conceptual graph using Searchable symmetric encryption

Sridharan .K [1], Shakthi.R [2], Revanth.R [3], Naveen.S [4], Shyam balaji.T [5]

[1]*Associate Professor,* [2345]*Students of B.Tech .Information technology,*
Panimalar engineering college, Chennai, Tamilnadu, India.

---***---

Abstract - *At present, explorable encipher is a hot topic in the field of cloud computing. The existing attainments are mainly concentrated on keyword-based search schemes, and almost all of them depend on predefined keywords extricated in the phases of index erection and query. However, keyword-based search schemes ignore the cognitive grammar representation information of users' repossession and cannot completely match users' search purpose. Therefore, how to design a content-based search scheme and make cognitive grammar search more effective and context-aware is a difficult challenge. In this System, for the first time, we define and solve the problems of cognitive grammar search based on conceptual graphs(CGs) over enciphered redistributed data in clouding computing (SSCG).We firstly employ the efficient measure of "sentence scoring" in text depiction and Stop words and stemming technique used to extract the most important and simplified topic sentences from documents. We then convert these simplified sentences into CGs. To perform measurable calculation of CGs, we design a new method that can map CGs to vectors. NER Algorithm is implemented for conceptual graph search, so that the algorithm used to systemize documents into five categories. Next, we executed TF/IDF algorithm to calculate the term frequency. Based on the term frequency, it produces the result to user. Then we have executed the HMAC Algorithm to find the resemblance between the documents. So the duplicate files will be separated.*

***Key Words*: Searchable encryption, cloud computing, smart semantic search, conceptual graphs, hmac, Named entity recognition.**

1. INTRODUCTION

NOWADAYS, a large number of data owners decide to store their individual data in the cloud which can help them attain the on-demand high-quality applications and services. It also reduces the cost of data management and storage facility spending. Due to the scalability and high efficiency of cloud servers, the way for public data access is much more scalable, low-cost and stable, especially for the small enterprises. However, data owners are puzzled by the privacy of data and existing schemes prefer to using data encryption to solve the problem of information leakage. How to realize an efficient searchable encryption scheme is a challenging and essential problem. Many existing recent schemes are keyword-based search including single keyword and multi-keywords etc. These schemes allow data users to retrieve interested files and return related documents in the encrypted form. However, due to connatural localization of keywords as document eigenvectors, the returned results are always imprecise and unable to satisfy intention of users. That means keywords as a document feature are inadequate data which carry relatively little semantic information. And some existing schemes hope to explore the relationships among keywords to expand the retrieval results. However, when extracting keywords from documents, the relationships among keywords are out of consideration which leads to the limitation of these schemes. So exploring a new knowledge representation with more semantic information compared with keywords to realize searchable encryption is a challenging and essential task. To solve the problem, we introduce Conceptual Graph (CG) as a knowledge representation tool In this System. CG is a structure for knowledge representation based on first logic. They are natural, simple and fine-grained semantic representations to depict texts. A CG is a finite, connected and bipartite graph.

2. PROPOSED SYSTEM

For the first time, we define the problem of semantic search based on conceptual graphs over encrypted outsourced data, and provide an effective method to perform a secure ranked search. Additionally, we define and present a round system of semantic search based on conceptual graphs over encrypted outsourced data.

Rather than employing traditional keywords as our knowledge representation tool, we use conceptual graphs to realize real semantic search in encrypted form. We propose a new practical and processing method of mapping CGs to vectors that makes quantitative calculation of CGs possible.

Extensive experimental results demonstrate the effectiveness and efficiency of the proposed solution. We proposed Named Entity recognition algorithm is used to classify the files into attributes like organization, location, date, person name etc.

Next, we implemented TF/IDF algorithm to calculate the term frequency. Based on the term frequency, it produces the result to user. Then we have implemented the HMAC Algorithm to find the similarity between the documents. So the duplicate files will be separated.

2.1 MODULE DESCRIPTION:

1. Data Owner

Data owner needs to login first and can upload the file to cloud and can view his own file. User Request for file download can be activated by data owner and sends key to the User. The data owner is responsible for collecting documents, building document index and outsourcing them in an encrypted format to the cloud server.

2. Data User

User needs to register before login by giving his/her own details. After Login, user sends request to the data owner for search. He/She can search in two ways

i)Normal Search

In Normal search, user can search by file name.

ii)Fine Search

In Fine search, User can search by the content inside the file like he/she can search by any words which is present in the file.

iii) Conceptual search

Here User can search by the attributes like organization, location, date, person name, etc. The File which is uploaded by the data owner is to classify the files into attributes like organisation, location, date, person name etc.

If user wants to download the file, he/she has to get authorization from the data owner. The data user needs to get the authorization from the data owner before accessing to the data. By using the attributes also User can search and download the required file.

3. Cloud Server

In this Module, The cloud can view both the data user and data owner informations. The Data which is uploaded by the data owner can be stored in the cloud.

3. ALGORITHM

3.1 STOP WORDS REMOVAL

A dictionary based approach is been utilized to remove stopword from document. A generic stopword list containing 75 stopwords created using hybrid approach is used. The algorithm is implemented as below given steps. The target text is tokenized and individual words are stored in array. A single stop word is read from stopword list. The stop word is compared to target text in form of array using sequential search technique. If it matches , the word in array is removed , and the comparison is continued till length of array. After removal of stopword completely, another stopword is read from stopword list and again algorithm runs continuously until all the stopwords are compared. Resultant text devoid of stopwords is displayed, also required statistics like stopword removed, no. of stopwords removed from target text, total count of words in target text, count of words in resultant text, individual stop word count found in target text is displayed.

3.2 NAMED ENTITY RECOGNITION

Named-entity recognition (NER) (also known as entity identification, entity chunking and entity extraction) is a subtask of information extraction that seeks to locate and classify named entities in text into pre-defined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages, etc.

ALGORITHM:

```
For i=1  to  n do //n:number of molecules
 W[i] =4;        //Start by the value of  width equal to four
a=Size[i]mod W [i];//size of molecule [i]
if(a=0)then
h[i] = size [i] div w[i];
Call function  verify (h[i],w[i]);
end if;
else
h[i] = (Size [i] div w[i]) + 1;
goto verify (h[i],w[i]);
write module  M[i](h[i]; w[i]);
Call  function KNER
Return the list of all NER
Place the module [i] in the first adequate NER start from the
left  to  right
If placement is found then Module accepted
End  if;
Else
Module  Rejected
end for;
```

3.3 TERM DOCUMENT FREQUENCY

First, the document frequency of each term in N and T is calculated accordingly. In the case of term set N, the document frequency of each term n is equal to the number of files (from dates d1 to d2) in which n has been selected as a keyword; it is represented as df(n). The document frequency of each term t in set T is calculated in a similar fashion.

3.4 DES ALGORITHM

DES key length and brute-force attacks. The Data Encryption Standard is a block cipher, meaning a cryptographic key and algorithm are applied to a block of data simultaneously rather than one bit at a time. To encrypt a plaintext message, DES groups it into 64-bit blocks

3.5 HMAC ALGORITHM

In cryptography, a keyed-hash message authentication code (HMAC) is a specific type of message authentication code (MAC) involving a cryptographic hash function and a secret cryptographic key. It may be used to simultaneously verify both the data integrity and the authentication of a message, as with any MAC. Any cryptographic hash function, such as MD5 or SHA-1, may be used in the calculation of an HMAC; the resulting MAC algorithm is termed HMAC-X, where X is the hash function used (e.g. HMAC-MD5 or HMAC-SHA1). The cryptographic strength of the HMAC depends upon the cryptographic strength of the underlying hash function, the size of its hash output, and the size and quality of the key.

ALGORITHM

Function hmac

Inputs

Key : Bytes array of bytes

message : Bytes array of bytes to be hashed

hash : Function the hash function to use (e.g. SHA-1)

blockSize: Integer the block size of the underlying hash function (e.g. 64 bytes for SHA-1)

outputSize: Integer the output size of the underlying hash function (e.g. 20 bytes for SHA-1)

Keys longer than blockSize are shortened by hashing them
if (length(key) > blockSize) then

key ← hash(key) //Key becomes outputSize bytes long

Keys shorter than blockSize are padded to blockSize by padding with zeros on the right

if (length(key) < blockSize) then

key ← Pad(key, blockSize) //pad key with zeros to make it blockSize bytes long

o_key_pad = key xor [0x5c * blockSize] //Outer padded key

i_key_pad = key xor [0x36 * blockSize] //Inner padded key

return hash(o_key_pad ∥ hash(i_key_pad ∥ message)) //Where ∥ is concatenation

5. ARCHITECTURE:

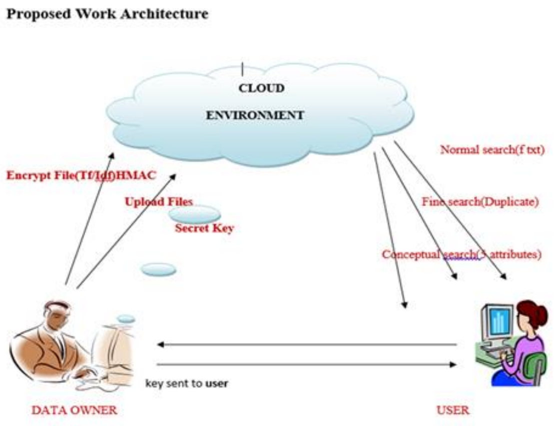

Fig -1: The architecture of ranked search over encrypted cloud data.

We summarize our system model showed in Fig. 1 which includes three entities: data owner, data user and cloud server.

1) Data Owner:

Data owner owns n data files F = {F1, F2,...,Fn}that he encrypts his source documents before they are outsourced to the cloud server. Also, he must guarantee that these documents can be searched effectively. In this System, the data owner encrypts their documents set and generates searchable indexes before outsourcing data to the cloud server. Besides this, the pre-process work such as the construction of CG, the transformation of CG into vectors and the update operation of documents should be handled ahead of time. The data user also should make a secure distribution of the key information of trapdoor generation and provide authorization for authorized data users.

2) Data Users

Data users should obtain a warrant from data owner to have access to documents. Data users should submit a simple sentence to generate a trapdoor and take back the documents which meet his requirement from the cloud server.

3) Cloud Server

Cloud server receives the store request from the data owner and execute the operation of storing the encrypted documents and searchable indexes. When the data users send the trapdoor to the cloud server, the cloud server makes a computation of relevance scores and returns top-k related documents to the data users. The cloud server is also responsible for executing the command of updating documents and searchable indexes.

6. CONCLUSIONS

In this System, compared with the previous study, we propose two more secure and efficient schemes to solve the problem of privacy-preserving smart semantic search based on conceptual graphs over encrypted outsourced data. Considering various semantic representation tools, we select Conceptual Graphs as our semantic carrier because of its excellent ability of expression and extension. To improve the accuracy of retrieval, we use Tregex simplify the key sentence and make it more generalizable. We transfer CG into its linear form with some modification creatively which makes quantitative calculation on CG and fuzzy retrieval in semantic level possible. We use different methods to generate indexes and construct two different schemes with two enhanced schemes respectively against two threat models by introducing the frame of MRSE. We implement our scheme on the real data set to prove its effectiveness and efficiency. For the further work, we will explore the possibility of semantic search over encrypted cloud data with natural language processing technology.

7. ACKNOWLEDGEMENT

A project of this magnitude and nature requires kind co-operation and support from many, for successful completion . We wish to express our sincere thanks to all those who were involved in the completion of this project.

8. REFERENCES

[1] S.Miranda-Jimnez, A.Gelbukh, and G.Sidorov, ”Summarizing conceptual graphs for automatic summarization task,” Conceptual Structures for STEM Research and Education,Springer Berlin Heidelberg,pp. 245-253,2013.

[2] R.Ferreira, L.de Souza Cabral, and R.D.Lins, ”Assessing sentence scoring techniques for extractive text summarization,” Expert systems with applications,vol.40,no.14,pp.5755-5764,2013.

[3] M.Liu, R.Calvo, and A.Aditomo, ”Using wikipedia and conceptual graph structures to generate questions for academic writing support,” Learning Technologies,IEEE Transactions on,vol.5,no.3,pp.251- 263,2012.

[4] M.Heilman, and N.A.Smith, ”Extracting simplified statements for factual question generation ,” Proceedings of QG2010: The Third Workshop on Question Generation,pp.11-20,2010.

[5]D.X.Song,D.Wagner,andA.Perrig,”Practicaltechniquesforsearches on encrypted data,” Proceedings of Security and Privacy,2000 IEEE Symposium on,pp.44-55,2000.

[6] Y.-C.Chang and M.Mitzenmacher, ”Privacy preserving keyword searches on remote encrypted data,” Proceedings of ACNS,pp.391421,2005.

[7] R.Curtmola, J.A.Garay, S.Kamara, and R.Ostrovsky, ”Searchable symmetric encryption: improved definitions and efficient constructions,” Proceedings of ACM CCS,pp.79-88,2006.

[8] C.Wang, N.Cao, and J.Li, ”Secure ranked keyword search over encrypted cloud data,” Proceedings of Distributed Computing Systems (ICDCS),2010 IEEE 30th International Conference on,pp.253-262,2010.

[9] N.Cao, C.Wang, and M.Li, ”Privacy-preserving multi-keyword ranked search over encrypted cloud data,” Parallel and Distributed Systems,IEEE Transactions on,vol.25,no.1,pp.222-233,2014.

[10] W.Sun, B.Wang, and N.Cao, ”Privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking,” Proceedings of the 8th ACM SIGSAC symposium on Information,computer and communications security,pp.71-82,2013.

[11] R.Li, Z.Xu, and W.Kang, ”Efficient multi-keyword ranked query over encrypted data in cloud computing,” Future Generation Computer Systems,vol.30,pp.179-190,2014.

[12] Z.Fu, X.Sun, and Q.Liu, ” Achieving Efficient Cloud Search Services: Multi-keyword Ranked Search over Encrypted Cloud Data Supporting Parallel Computing,” IEICE Transactions on Communications, vol.98,no.1,pp.190-200,2015.

[13] Z.Xia, X.Wang, and X.Sun, ”A Secure and Dynamic Multikeyword Ranked Search Scheme over Encrypted Cloud Data,”Parallel and Distributed Systems,IEEE Transactions on, vo.27,no.2,pp.340-352,2015.

[14] Z.Fu, J.Shu, and X.Sun, ”Semantic keyword search based on trie over encrypted cloud data,” Proceedings of the 2nd international workshop on Security in cloud computing,ACM, pp.59-62,2014.

[15] J.F.Sowa, ”Conceptual structures: information processing in mind and machine,” 1983.