ISSN: 0974-2115
www.jchps.com Journal of Chemical and Pharmaceutical Sciences

An efficient multi keyword ranked search over encrypted cloud supporting semantic query

J.Visumathi[1], A.Christina[2]
[1]Department of CSE, Jeppiaar Engineering College, Tamil Nadu, India.
[2]Assistant Project Consultant, TCS, Chennai.
***Corresponding author: E-Mail:**jsvisu@gmail.com

ABSTRACT

With the advent of cloud computing, data owners are motivated to outsource their complex data from local sites to commercially available public cloud. But for protecting sensitive information like military information, personal health documents etc, these documents have to be encrypted before outsourcing; which prevents traditional search methodologies using plain text keyword search. Considering the ever growing number of data owners and documents in cloud, it is necessary to allow multi keywords in search requests and return documents in the order of their relevance to keywords. Meanwhile existing approaches only support fuzzy keyword based search but not semantic based search. There are works based on multi keyword ranked search using synonymous queries but not based on semantics. For securing the data in the cloud, we use (Advanced Encryption Standard) AES algorithm to encrypt the data and (Reverse Advanced Encryption Standard) RAES to decrypt the same. We use secured (k-Nearest Neighbour) KNN algorithm to achieve secured search. In order to provide semantic based search we include a tool called wordnet where we perform semantic based analysis of the files and then include secured KNN over the documents and rank them according to their order of relevance primarily using the term and inverse document frequency.

KEY WORDS: Cloud Computing, Searchable Encryption, Privacy-preserving, Multiple Keyword Search, Relevance Ranking, Semantic Query.

1. INTRODUCTION

Due to the rapid expansion of data, data owners tend to store their data into the cloud to release the burden of data storage and maintenance as shown in fig 1. However, as the cloud customers and the cloud server are not in the same trusted domain, our outsourced data maybe exposed to risk. Thus, before being sent to cloud, the sensitive data must be encrypted. As the data is being encrypted, the server cannot perform normal text based search. Thus plaintext keyword based search cannot be directly applied to the encrypted cloud data. Traditional methods of Information Retrieval are present that provides the user with multi keyword ranked search but no encryption of data is done.

Similarly cloud needs to provide data users with efficient search mechanisms, while protecting data and search privacy. Meaningful data storage to cloud is only possible if it can be easily retrieved and protected. We have related works in which data owners build a searchable inverted index that stores a list of mapping from keywords to the corresponding list of files that has the particular k-word. When the data owner inputs a keyword, a trapdoor is generated for this k-word and then submitted to the server. Upon receiving the trapdoor the cloud server performs comparison between k-words and index files and returns all the files containing the k-word but cloud computing is a 'pay as you go' network ; so in order to zero in on the exact data file user may use multiple key words and in order to provide the user with most pertinent information the cloud server ranks the data files in the order of relevance as entered by the user,but still the user may also search for related documents or related information; this necessitates semantic based search. For efficient data utilization and security we use semantic based multi keyword ranked search. Among various multi-keyword semantics, we choose "coordinate matching" i.e, as many matches as possible. We further use "inner product similarity" to quantitatively evaluate such similarity measure.

Figure.1. Cloud computing- various application

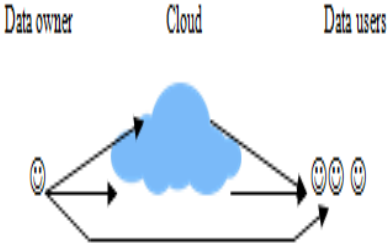

Figure.2. (Search Control) Encrypted queries, (Access control) Sharing of key

Coordinate Matching: It is an intermediate similarity measure which uses the number of query key words appearing in the document to quantify the relevance of that document. Based on the similarity measure it provides as many documents as possible.

Inner Product Similarity: Used for the effective identification of the document. There is a data vector and query vector and we find the product of the two vectors both vectors; based on the dot product we come to the conclusion whether it is relevant or not.

The proposed scheme solves the multi key word search, by using latent semantic analysis and ranked search,we get documents having similar meaning to the(Key word) k-word for e.g, if you type car as the keyword you would also get the files with automobiles as the keyword as output, as car and automobiles are semantically correct.

Figure.3. Secured Cloud

Related Work: Fu (2014), proposed multiple keyword based search over encrypted cloud data and ranking the same according to relevance and also supporting synonym based query, but it doesn't give an idea to implement semantic based queries.

Ning Cao (2014), proposed how Multi Keyword Ranked Search is to be done in Encrypted cloud and rank the same according to the relevance of the documents. But the paper does not support Synonym or semantic based query.

Kamara (2010), provided an overview about storing of encrypted data in cloud. It covers the likes of efficient data storage but it does not account for data retrieval methodologies

Singhal (2001), proposed retrieval of data in modern systems. The paper gives immense details about data recovery but efficient search methods are not covered in the paper

Song (2000), proposed practical methods for searching on encrypted data but it does not give an idea about searching encrypted data in cloud and about privacy.

Goh (2003), proposed information regarding secured indices for retrieval of stored data. But the work does not give idea about encryption of data in cloud and secured storage

Cao (2011), proposed privacy based query over encrypted cloud data which is in graphical data. But the work does not cover storage and search for text documents.

MRSE (Multi Keyword Ranked Search over Encrypted Cloud) uses known cipher text model; where the cloud only knows the encrypted data set and searchable index i. Data and key are encrypted using AES algorithm they are separately stored in cloud.

2. MATERIALS AND METHODS

Motivation For Proposed System: We have so far looked the aspects of a Multi Key Word Ranked Search supporting Synonymous query, but sometimes the user may want just not the exact file but also some other related documents pertaining to the query word given so we put forth a need for a semantic based search on the encrypted cloud, analyzing, encrypting and providing a secured semantic based search on cloud data is simple and provides effective search on to the next level, thus helping the users and the owners of the documents of the cloud.

The tools employed for providing semantic based search are wordnet the dictionary tool having all the semantic meaning of the key word provided, we connect this tool to the data owner end as well as the data user end. As the data owner outsources his files into the cloud, the data before being stored into the cloud is encrypted using AES algorithm using a private key, the encrypted data is then stored into the cloud, the data user searches for the relevant document using (Latent Semantic Analysis) LSA, wordnet tool, KNN, (Inverse Term Document Frequency) ITDF and TF, after he zeroes-in one of the documents, he uses this private key to decrypt the data for which we use RAES.

System Architecture

Data Owner End: In the data owner end, he has the set of files (f1, f2,..fn) to be stored on the cloud but before this the data is to be encrypted, this is performed by using AES algorithm on the data to be uploaded. The data owner has a private key that is to be shared with the authorized cloud users only.

Preprocessing End: In this part we perform the generation of the keywords, for this we use a stemmer that eliminates all the verbs, punctuations and stop words. Here we find the document term frequency and also the semantic meaning of the keywords for search. Then we encrypt the data as well as the searchable index and then store it in cloud. A document matrix and inverse document matrices are created as well.

Cloud Server: Cloud stores the encrypted data and searchable index. The search operations are performed on the encrypted data by the cloud, it performs secured search using secured KNN algorithm. Rank is given based on term frequency and inverse document frequency, a binary tree is created based on the order of relevance. That is ranked according to the order of relevance of the document.

Data User End: The data user puts in the keywords for searchable query, meanwhile the semantic meaning for the query keywords are found using Wordnet tool, all these are encrypted and send to the cloud, cloud performs the searching operations and send the relevant data to the user, the user decrypts the same using the key provided by the data owner, using RAES algorithm.

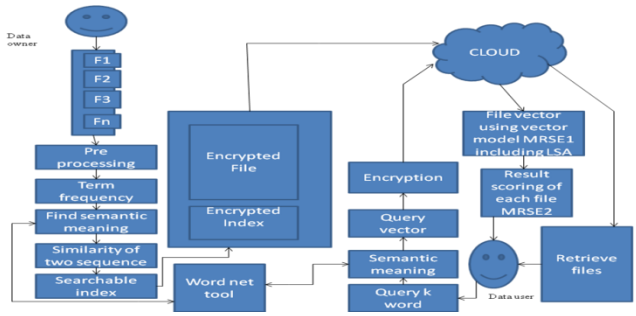

Figure.4. Architecture Diagram of the Proposed System

Proposed System: We modify the existing system to include semantic search. From the retrieval point, it modifies the search algorithm by introducing Word Net tool and advanced KNN algorithm.

Latent Semantic Analysis: It is an indexing and retrieval method; where words are identified on semantic basis. It is used to identify similarity patterns in a collection of text. It assumes that words used in similar context will have same meaning. Key feature is its ability to retrieve conceptual content of a body of text by establishing association between the terms used in similar contexts. Mathematics behind LSA is that we construct a rectangular Matrix of words from the file, each tuple containing the transform of the number of times a particular word appears in the passage.

After constructing the document matrix we apply TF weight over it. On the data owner side, then owner constructs an index and a sub-index for data document as discussed in the existing paper, in addition to it we introduce a semantic library called wordnet, when he adds on a k-word the semantic meaning for the same is searched and introduced to the index terms, and then the keywords, their semantic meaning are both encrypted.

The cloud server searches the files based on the query and ranks them using KNN Algorithm; we construct a document tree based on the rank of the file.

Notation:

- *f*- the plain text document collection, denoted as asset of m-documents
- *f* - (f1,f2…fm)
- *c* - the encrypted cloud document stored in the cloud server, denoted as
- *c* -(c1,c2…cm)
- *w* - the dictionary, i.e, the keyword set consisting n keywords, denoted as
- *w* - (W1, W2….Wm)
- *I* - searchable index with semantics associated with c, denoted as *(I1,I2,…Im*) where each sub index Ii is built for fi.
- *S* - semantic meaning for each index I,
- Denoted as (s1, s2…sm) included in the index.
- w_i - subset of *W*, representing the keywords in a search request, denoted as
- *wi-(Wi1,Wi2,..Wik)*
- f_w - the ranked id list of all documents according to *Wi*

Semantic Expansion: Semantic words are the words with the same meaning or the words that are used in the same context. Latent Semantic Analysis takes that the words that are used in the same context has similar meaning. A common semantic thesaurus is built on the foundation of Princeton Universitie's Wordnet tool.

Rank Function: In, information retrieval rank function is used for ranking the documents according to the order of their relevance. We use Term Frequency (TF) and Inverse Document Frequency (IDF) for ranking the documents.TF

represents the key terms in the document, whereas IDF is a quantitative measure of files containing the keywords and the total number of files in the private cloud.

- The similarity function is obtained from : $$S(Q,D)=\frac{\sum_{j=1}^{n} wq,j \,.\, wd,j}{\sqrt{\sum_{j=1}^{n}(wq,j)^2}\,.\,\sqrt{\sum_{j=1}^{n}(wd,j)^2}}$$

Where $w_{d,j} = 1+\ln f_{d,j}$, $w_{q,j} = \ln(1+\frac{N}{fj})$

TF and IDF are $\sqrt{\sum_{j=1}^{n}(wq,j)^2}$ and $\sqrt{\sum_{j=1}^{n}(wd,j)^2}$

Where,

- $f_{d,j}$ - the TF of keyword w_j within the document d.
- f_j -the number of documents containig the keyword w_j
- M - total number of documents in the document collection.
- N -total number of keywords in the keyword dictionary
- $w_{d,j}$ - TF computed for $f_{d,j}$
- $w_{q,j}$ - IDF computed from N and f_j

Construction Of Keywords Extended By Semantics: Keywords are to be extracted firstly from cloud before outsourcing. TFIDF (Term Frequency-Inverse Document Frequency) is used to extract the k-words.

W_{ik}= TF*IDF=TF*$\frac{1}{DF}$ = f_{ik}* $\frac{\log N}{n_k}$

F_{ik} is the frequency of term I in a text N, n_k is the total number of texts which contains i.

In order to provide a better semantics based analysis, the k-word needs to be extended by semantic meaning. For which WordNet tool from Princeton University is used.

1. Selecting the synonyms
2. Selecting the words which can be semantically substituted

Now the keyword set with semantics will be:

File 1 : k_{f1}^1 or s^1,…..k_{f1}^nor s^n

File m: k_{fm}^1ors^1_m,….k_{fm}^nor s^n_m

Secured KNN: In order to compute secured inner product similarity in a privacy preserving environment we use the secured K-nearest neighbour algorithm (Wong, 2009). This method uses the distance between the databases and query to find the nearest neighbor to the query point. We represent a data vector as p and query vector as q, secret key is composed of one n bit vector as x and two n*n matrices

As $\{M_1,M_2\}$.q, the data and query vectors are split into two random vectors as $\{p',p''\}$.q and $\{q',q''\}$.p, here x vector acts as a splitting vector

If jth bit of x is 0, then p'[j] and p"[j] are set same as p[j], while q'[j] and q"[j] are assigned some random values so that their sum is equal to q[j].; if jth bit is zero the assignment is vice-versa. The split data vector is encrypted as$\{M^T_1.$ P',M^T_2. P"} and query vector $\{M^{-T}_1.$ q',M^{-T}_2. q"}.

The score is calculated as:

$\{M^T_1.$ P',M^T_2. P"}. $\{M^{-T}_1.$ q',M^{-T}_2. q"}

= $M^T_1.$ P',M^T_2. P"+$M^{-T}_1.$ q',M^{-T}_2. q"

= p'^T.q'+p''^T.q" = p'^T.q'; Clearly the score is not affected by encryption. Without prior knowledge neither the query vector nor the data vector can be guessed by the cloud.

Latent Semantic Analysis During Information Retrieval: In information retrieval, LSA is used for finding the semantic relationship; It uses a mathematical function called (Singular Value Decomposition) SVD to find semantic structure between terms and documents. Term document matrix is constructed consisting n rows each of which represents a data vector for each file.

A'=(A'[1]….A'[j]…..A'[n])=

$$\begin{matrix} TF_{1,1} & TF_{1,2} \dots & TF_{1,m} \\ TF_{n-1,1} & TF_{n-1,2} \_ & TF_{n-1,m} \\ TF_{n,1} & TF_{n,2} \_ & TF_{n,m} \end{matrix}$$

We take this large term document matrix and decompose it to a set of k, orthogonal factors from which the original matrix can be estimated. For example a term document matrix A can be decomposed to three other matrices

A= M'.L'.N^T'

M' is an m by r term-concept matrix, S' is an r by r diagonal matrix and $N^{T'}$ is an n by r concept-document matrix. Thus this splitting of matrix A into three distinct matrices is called singular value decomposition of A. Here m is the number of unique terms in the document and n is the number of documents and r is the rank of A. The product of the resulting matrices is a matrix A which is approximately equal to matrix A' i.e, A= M'.L'.$N^{T'}$ ≅ A'.

3. RESULT AND DISCUSSION

Graph For Index Tree Construction: The graph describes the time for the generation of document tree based on the query key word given, it is found that there is a steady increase in the number of trees generated.

Graph for Search Time: It shows the performance of the proposed scheme in relation to the search time taken. The graph shows a steady increase in search time as the number of dataset is increased they have a linear relationship.

Graph for Index Storage Cost: The graph gives the storage cost for index terms with respect to the number of keywords, they also show a linear relationship

Graph for Measure of Relevance: It is the measure of the relevance of the documents provided by the server after processing the query. The cloud should rank the documents based on the order of relevance of the query key word that the user has provided. It has been found that our method provides remarkable results in the retrieval of documents based on the order of their relevance.

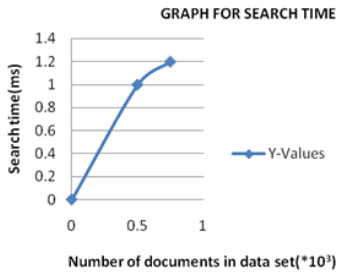

Figure.6. Search Time

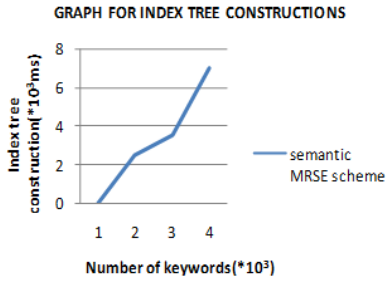

Figure.5. Index Tree Construction Time

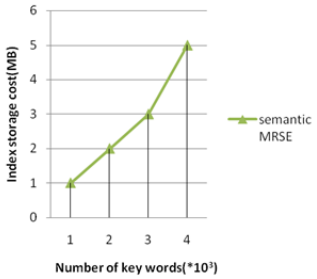

Figure.7. Index Storage Cost

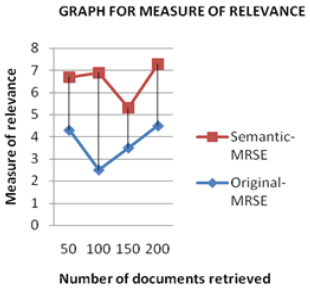

Figure.8. Measure of Relevance

4. CONCLUSION AND FUTURE WORK

In this paper, a multi-keyword ranked search over encrypted cloud data is proposed, which supports Latent Semantic Indexing which we use for semantic keyword generation. We construct term document matrices from which we analyze semantic relationship between different documents, we use a secured KNN Algorithm to enable privacy based ranking and document matrix tree for privacy based search, so that we can not only get the exact files but also perform ranking and search in a secured manner. The proposed system does not only gives the exact matching file but also files with that are semantically associated to the query. Future works include storing image and video files as well as applying MRSE with semantics for efficient data retrieval.

REFERENCES

Brinkman, Searching in Encrypted Data, PhD thesis, Univ, of Twente, 2007.

Cao N, Effective Cloud Services- Multi Keyword Ranked Search, 2014.

Cao N, Yang Z, Wang C, Ren K and Lou W, Privacypreser- ving Query over Encrypted Graph-Structured Data in Cloud Computing, Proc. Distributed Computing Systems (ICDCS), 2011, 393-402.

Cao S, Yu Z, Yang W, Lou and Hou Y, LT Codes-Based Secure and Reliable Cloud Storage Service, Proc. IEEE INFOCOM, 2012, 693-701.

Goh E.J, Secure Indexes, Cryptology e Print Archive, 2003.

Kamara S and Lauter K, Cryptographic Cloud Storage, Proc.14th Int'l Conf. Financial Cryptograpy and Data Security, 2010.

Lewko A, Okamoto T, Sahai A, Takashima K and Waters B, Fully Secure Functional Encryption: Attribute-Based Encryption and (Hierarchical) Inner Product Encryption, Proc. 29th Ann. Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT '10), 2010.

Li M, Yu S, Cao N and Lou W, Authorized Private Keyword Search over Encrypted Data in Cloud Computing, Proc. 31st Int'l Conf. Distributed Computing Systems (ICDCS '10), 2011, 383- 392.

Singhal A, Modern Information Retrieval, A Brief Overview, IEEE Data Eng. Bull, vol. 24(4), 2001, 35-43,

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

Wang C, Cao N, Ren K, and Lou E, Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data, IEEE Trans. Parallel and Distributed Systems, vol. 23(8),2012, 1467- 1479.

Wang C, Wang Q, Ren K and Lou W, Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing, Proc. IEEE INFOCOM, 2010

Witten IH, Moffat A, and Bell T.C, Managing Gigabytes, Compressing and Indexing Documents and Images, Morgan Kaufmann Publishing, 1999.

Wong WK, Cheung DW, Kao B and Mamoulis N, Secure kNN Computation on Encrypted Databases, Proc. 35th ACM SIGMOD, Int'l Conf. Management of Data, (SIGMOD), 2009, 139-152.

Zhangi fu, Xingmin sun, Effective Cloud Services MRSE using Synonymous Query, IEEE transactions on consumer electronics, 2014