ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 1, January 2014

Multikeyword based approach for retrieving the Encrypted Cloud Data

S.Thamizharasan[1], P.Mahalakshmi[2], V.Vijayalakshmi[3]

Student M.Tech, CSE Department, Christ College of engineering and technology, Pondicherry, India[1]

Student M.Tech, CSE Department, Christ College of engineering and technology, Pondicherry, India [2]

Assistant Professor, CSE Department, Christ College of engineering and technology, Pondicherry, India [3]

Abstract: Data can be stored and retrieved anywhere and anytime with the help of the cloud. But retrieving the encrypted data is a challenging one for the user, and the later keyword based approach overcomes that problem. Even though the keyword based approach had arrived customer satisfaction is only little in that approach. So a new technique known as the Multikeyword based approach in searching and retrieving the encrypted cloud data full fills the customer satisfaction. The Multikeyword based approach can be implemented with the model named as a Vector Space model. This model can be used to retrieve the encrypted cloud data in an efficient manner and the customer satisfaction is fulfilled.

Keywords: Rank, Encrypted Data, Retrieve, Multikeyword

I. INTRODUCTION

The cloud makes it possible for you to access your information from anywhere at any time. While a traditional computer setup requires you to be in the same location as your data storage device, the cloud takes away that step. The cloud removes the need for you to be in the same physical location as the hardware that stores your data. Your cloud provider can both own and house the hardware and software necessary to run your home or business applications. This is especially helpful for businesses that cannot afford the same amount of hardware and storage space as a bigger company. Small companies can store their information in the cloud, removing the cost of purchasing and storing memory devices.

Additionally, because you only need to buy the amount of storage space you will use, a business can purchase more space or reduce their subscription as their business grows or as they find they need less storage space. Each provider serves a specific function, giving users more or less control over their cloud depending on the type. When you choose a provider, compare your needs to the cloud services available. The information housed on the cloud is often seen as valuable to individual with malicious intent. There is a lot of personal information and potentially secure data that people store on their computers, and this information is now being transferred to the cloud. This makes it critical for you to understand the security measures that your cloud provider has in place, and it is equally important to take personal precautions to secure your data. The multikeyword retrieval over encrypted cloud data achieves high security and privacy.

II. RELATED WORK

As Cloud Computing becomes prevalent, sensitive information are being increasingly centralized into the cloud. For the protection of data privacy, sensitive data has to be encrypted before outsourcing, which makes effective data utilization a very challenging task. Ranked search greatly enhances system usability by returning the matching files in a ranked order regarding to certain relevance criteria (e.g., keyword frequency), thus making one step closer towards practical deployment of privacy-preserving data hosting services in Cloud Computing. We first give a straightforward yet ideal construction of ranked keyword search under the state-of-the-art searchable symmetric encryption (SSE)[1].By determining the most common English words and phrases since the beginning of the sixteenth century, we obtain a unique large-scale view of the evolution of written text. We find that the most common words and phrases in any given year had a much shorter popularity lifespan in the sixteenth century than they had in the twentieth century [2]. The task of compiling a monograph on corpus linguistics must be faced up with the problem that there is at present no consensus among linguists. It is generally admitted that the mere fact of quoting authentic examples of language use is not a sufficient condition for a piece of research to qualify' as corpus linguistic [3]. they provide provable secrecy for encryption, in the sense that the untrusted server cannot

learn anything about the plaintext when only given the cipher text; they provide query isolation for searches, meaning that the untrusted server cannot learn anything more about the plaintext than the search result; they provide controlled searching, so that the untrusted server cannot search for an arbitrary word without the user's authorization; they also support hidden queries, so that the user may ask

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 1, January 2014

the untrusted server to search for a secret word without revealing the word to the server [4].

Privacy preserving multi-keyword ranked search over encrypted cloud data (MRSE). We establish a set of strict privacy requirements for such a secure cloud data utilization system. Among various multikeyword semantics, we choose the efficient similarity measure of "coordinate matching", i.e., as many matches as possible, to capture the relevance of data documents to the search query. We further use "inner product similarity" to quantitatively evaluate such similarity measure [5]. Fuzzy keyword search [6] greatly enhances system usability by returning the matching files when users' searching inputs exactly match the predefined keywords or the closest possible matching files based on keyword similarity semantics, when exact match fails. In our solution, we exploited it distance to quantify keywords similarity and develop two advanced techniques on constructing fuzzy keyword sets, which achieve optimized storage and representation overheads. A new symbol-based tire-traverse searching scheme, where a multi-way tree structure is built up using symbols transformed from the resulted fuzzy keyword sets is constructed. If the user is actually interested in documents containing each of several keywords the user must either give the server capabilities for each of the keywords individually or rely on an intersection calculation to determine the correct set of documents, or alternatively, the user may store additional information on the server to facilitate such searches. Neither solution is desirable; the former enables the server to learn which documents match each individual keyword of the conjunctive search and the latter results in exponential storage if the user allows for searches on every set of keywords[7] [8].

III. EXISTING SYSTEM

A series of searchable symmetric encryption (SSE) schemes have been proposed to enable search on ciphertext. Traditional SSE schemes enable users to securely retrieve the ciphertext, but these schemes support only Boolean keyword search, i.e., whether a keyword exists in a file or not, without considering the difference of relevance with the queried keyword of these files in the result. So that data users have to know specific terms associated with the search topic in order to get useful results.

IV. PROPOSED WORK

Vector space model or term vector model is an algebraic model for representing text documents as vectors of identifiers, such as, for example, index terms. It is used in information filtering, information retrieval, indexing and relevancy rankings. The vector space model procedure can be divided in to three stages. The first stage is the document indexing where content bearing terms are extracted from the document text. The second stage is the weighting of the indexed terms to enhance retrieval of document relevant to the user. The last stage ranks the document with respect to the query according to a similarity measure.

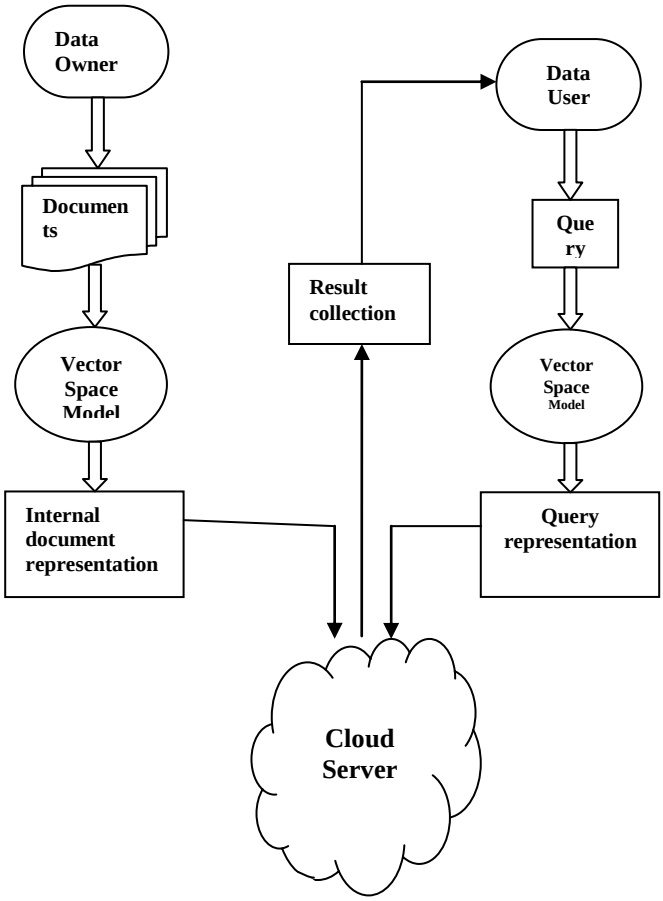

Fig. 1 Proposed Architecture

A. Document Indexing

It is obvious that many of the words in a document do not describe the content, words like the, is. By using automatic document indexing those non significant words (function words) are removed from the document vector, so the document will only be represented by content bearing words. This indexing can be based on term frequency, where terms that have both high and low frequency within a document are considered to be function words. In practice, term frequency has been difficult to implement in automatic indexing. Instead the use of a stop list which holds common words to remove high frequency words (stop words), which makes the indexing method language dependent. In general, 40-50% of the total numbers of words in a document are removed with the help of a stop list.

Non linguistic methods for indexing have also been implemented. Probabilistic indexing is based on the assumption that there is some statistical difference in the distribution of content bearing words, and function words. Probabilistic indexing ranks the terms in the collection. The term frequency in the whole collection. The function words

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 1, January 2014

are modelled by a Poisson distribution over all documents, as content bearing terms cannot be modelled. The use of Poisson model has been expand to Bernoulli model. Recently, an automatic indexing method which uses serial clustering of words in text has been introduced. The value of such clustering is an indicator if the word is content bearing.

B. Term Weighing

Term weighting has been explained by controlling the exhaustively and specificity of the search, where the exhaustively is related to recall and specificity to precision. The term weighting for the vector space model has entirely been based on single term statistics. There are three main factors term weighting: term frequency factor, collection frequency factor and length normalization factor. These three factor are multiplied together to make the resulting term weight. The term frequency is somewhat content descriptive for the documents and is generally used as the basis of a weighted document vector. It is also possible to use binary document vector, but the results have not been as good compared to term frequency when using the vector space model.

There are used various weighting schemes to discriminate one document from the other. In general this factor is called collection frequency document. Most of them, e.g. the inverse document frequency, assume that the importance of a term is proportional with the number of document the term appears in. Experimentally it has been shown that these document discrimination factors lead to a more effective retrieval, i.e., an improvement in precision and recall.

The third possible weighting factor is a document length normalization factor. Long documents have usually a much larger term set than short documents, which makes long documents more likely to be retrieved than short documents. Different weight schemes have been investigated and the best results, recall and precision, are obtained by using term frequency with inverse document frequency and length normalization

C. Similarity Coefficients

The similarity in vector space models is determined by using associative coefficients based on the inner product of the document vector and query vector, where word overlap indicates similarity. The inner product is usually normalized. The most popular similarity measure is the cosine coefficient, which measures the angle between the document vector and the query vector.

D. Score Calculate

Scoring is a natural way to weight the relevance. Based on the relevance score, files can then be ranked in either ascending or descending. Several models have been proposed to score and rank files in IR community. Among these schemes, we adopt the most widely used one tf-idf weighting. The tf-idf weighting involves two attributes: Term frequency and inverse document frequency.

1) Inverse document frequency: A mechanism for attenuating the effect of terms that occur too often in the collection to be meaningful for relevance determination. An immediate idea is to scale down the term weights of terms with high *collection frequency*, defined to be the total number of occurrences of a term in the collection. The idea would be to reduce the tf weight of a term by a factor that grows with its collection frequency. Instead, it is more commonplace to use for this purpose the document frequency f_t, defined to be the number of documents in the collection that contain a term t. This is because in trying to discriminate between documents for the purpose of scoring it is better to use a document-level statistic than to use a collection-wide statistic for the term**.**

The reason to prefer df to cf is the collection frequency (cf) and document frequency (df) can behave rather differently. In particular, the cf values for both try and insurance are roughly equal, but their df values differ significantly. Intuitively, we want the few documents that contain insurance to get a higher boost for a query on insurance than the many documents containing try get from a query on try. Denoting as usual the total number of documents in a collection by N, we define the inverse document frequency of a term t as follows

$$\text{idf}_t = \log \frac{N}{\text{df}_t}.$$

2) Tf- idf weighting: It is the combination of term frequency and inverse document frequency, to produce a composite weight for each term in each document. The tf-idf weighting scheme assigns to term t a weight in document d given by

$$\text{tf-idf}_{t,d} = \text{tf}_{t,d} \times \text{idf}_t.$$

In other $\text{tf-idf}_{t,}$ ds, assigns to term t a weight in document d that is

- Highest when t occurs many times within a small number of documents (thus lending high discriminating power to those documents)
- Lower when the term occurs fewer times in a document, or occurs in many documents (thus offering a less pronounced relevance signal);
- Lowest when the term occurs in virtually all documents.

V. CONCLUSIONS

Retrieving the encrypted cloud data based on the customer needs is the challenging one, and also the retrieved data does not fulfil the customer. In this paper we use the Vector Space to retrieve the encrypted data from the cloud based on the Scoring. Scoring is a natural way to weight the

ISSN (Online) : 2278-1021
ISSN (Print) : 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 3, Issue 1, January 2014

relevance. Based on the relevance score, files can then be ranked in either ascending or descending and it is retrieved accordingly. It has the ability to incorporate term weights, measure similarities between almost anything such as ranking documents according to their possible relevance. So with this model the customer satisfaction and the efficient retrieval are possible without affecting the privacy of the data.

REFERENCES

[1] C. Wang, N. Cao, J. Li, K. Ren, and W. Lou, "Secure Ranked Keyword Search over Encrypted Cloud Data," Proc. IEEE 30th Int'l Conf. Distributed Computing Systems (ICDCS),2010.
[2] M. Perc, Evolution of the Most Common English Words and Phrases over the Centuries, J. Royal Soc. Interface,2012.
[3] S. Gries, Useful Statistics for Corpus Linguistics A Mosaic of Corpus Linguistics: Selected Approaches,Aquilino Sanchez Moises Almela, eds., pp. 269-291, Peter Lang, 2010.
[4] A. Swaminathan, Y. Mao, G.-M. Su, H. Gou, A.L. Varna, S. He, M. Wu, and D.W. Oard, Confidentiality-Preserving Rank-Ordered Search, Proc. Workshop Storage Security and Survivability,2007.
[5] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-Preserving Multikeyword Ranked Search over Encrypted Cloud Data," Proc. IEEE INFOCOM,2011.
[6] Fuzzy keyword search over encrypted data in cloud computing World Journal of Science and Technology 2012, 2(10):177-185
[7] P. Golle, J. Staddon, and B. Waters, Secure Conjunctive Keyword Search over Encrypted Data, Proc. Second Int'l Conf. Applied Cryptography and Network Security (ACNS),pp. 31-45, 2004.
[8] L. Ballard, S. Kamara, and F. Monrose, Achieving Efficient Conjunctive Keyword Searches over Encrypted Data,, Proc. Seventh Int'l Conf. Information and Communications Security (ICICS), 2005.