Research Article

Efficient Multi Keyword Search And Retrieve Encrypted Cloud Data Using Iddfs And Fpgrowth Algorithm

[1]D. Pradeepa, [2]Dr. P. Sumathi.

[1]Research Scholar, PG & Research Department of Computer Science, Government Arts College, Coimbatore.
[2]Assistant Professor, PG & Research Department of Computer Science, Government Arts College, Coimbatore.

Article History: Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 23 May 2021

Abstract - Cloud computing is a new technology that is paving the way for a plethora of new technologies and possibilities. The cloud's use has skyrocketed as a result of the unending advantages it provides. This increase in cloud use encouraged end-users to outsource their documents to the cloud, thus reducing financial burdens. Outsourcing sensitive information to a third-party server is often a risk since the data can be accessed by malicious insiders or outsiders in its original form. Researchers have introduced various schemes for searching outsourced data, but there is often a tradeoff between search efficiency and search capabilities of these searchable encryption schemes. To maintain a balance between search efficiency and search capabilities, a hybrid search scheme that can provide relevant results while preserving search efficiency is needed. The hybrid encrypted multi-keyword search scheme was introduced in this paper. The proposed scheme first conducts a conjunctive keyword search, in which documents containing the entire query words are easily retrieved and sorted based on their relevance to the query. If the number of obtained results exceeds the number of desired results, the search is efficient. If, on the other hand, the results fall below the end user-specified threshold, disjunctive searching is used. The records containing the subset of query words are included in the results of disjunctive searching. The obtained results are rated and shared with the end users. The search efficiency, search accuracy (measured in terms of recall and precision), and rank efficiency are all analyzed.

Keywords: - Cloud Storage, Cloud User, Multi-Keyword Search, Information Retrieval, Security.

1 INTRODUCTION

Cloud computing allows new forms of services through which computational and network resources are accessible via the Internet. Data outsourcing is one of the most common cloud computing services. Simultaneously, the security of remotely stored data on an untrusted cloud server is a major concern. Sensitive data, such as personal health records, addresses, income tax and financial reports, and so on, are usually outsourced in encrypted form using well-known cryptographic techniques to alleviate these concerns. While encrypted data storage protects remote data from unauthorized access, it complicates some simple yet necessary data utilization services such as plaintext keyword search. A simple solution of uploading the data, decrypting it, and searching locally is obviously impractical, since storing data in the cloud is useless unless it can be easily searched and used. Given the potentially large number of on-demand data users and massive amount of outsourced data documents in the cloud, this issue is especially difficult to solve because it is incredibly difficult to satisfy the requirements of accuracy, system accessibility, and scalability all at the same time.

The similarity search problem is made up of a set of data items that are distinguished by certain attributes, a query that specifies a meaning for a specific feature, and a similarity metric that measures the relevance between the query and the data items. However, these techniques either do not allow for multiple keyword searches and rating retrieved documents based on similarity ratings, or they are extremely computationally intensive. Furthermore, none of the similarity search schemes are immune to an adaptive adversary [5] that considers the history of the cloud user's queries as well as the collection of corresponding documents returned to the cloud user.

In this paper, we propose a hybrid encrypted multi-keyword search that returns matching data items in ranked order. The proposed protected searchable encryption scheme allows for multi-keyword queries over an encrypted document corpus and retrieves related documents ranked according to a similarity ranking. Unlike all previous schemes, our search is sub-linear to the total number of documents that contain the queried set of keywords. Create a tree-based index structure and propose an "Iterative Deepening - Depth-first Search (IDDFS)" algorithm based on it to achieve high search performance. Because of the unique structure of our tree-based database, the proposed search scheme will achieve sub-linear search times and deal with record deletion and insertion. The proposed EHRMS scheme uses the FPgrowth algorithm to find frequent Itemset in a transaction database without generating candidates. The FPgrowth algorithm represents the database as a tree known as a frequent pattern tree or FP tree. The relationship between the item sets will be maintained by this tree structure.

The rest of the paper is organized as follows: Section 2 gives an outline of the most recent related work. In Section 3, provide the framework of the proposed searchable encryption scheme and define the necessary security terms and requirements. The experimental findings and comparisons are discussed in section 4. Finally, in section 5, the concluding remarks are illustrated.

2. RELATED WORK

A systematic review is useful for research activities because it summarizes current techniques related to a research topic and recommends future research directions. There has been a lot of research performed on single-keyword and multi-keyword searches for encrypted data in the cloud. Traditional searchable encryption

schemes typically construct an encrypted searchable index whose content is protected from the server while still allowing document searching with a given search query. The aim of this review is to compare existing searchable encryption solutions and recognize their shortcomings through a systematic assessment.

Sun, J., et al. [2019] propose a novel powerful ranked multi-keyword retrieval scheme with keyword privacy for multiple data owners, allowing the cloud server to conduct multi-keyword searches over cloud data and then return the top-k ranked search results to data users without leaking any keyword or trapdoor information Several research works on graded multi-keyword search over encrypted cloud data under multiple data owner model have been established to address privacy concerns. Most of these systems, however, are vulnerable to keyword guessing and equivalence test attacks. Finally, the performance assessment shows that our scheme outperforms the current graded multi-keyword search schemes in terms of functionality.

Zhang, W., et al. [2014] proposed in a multi-owner model, a scheme for dealing with stable ranked multi-keyword search several studies under the single owner model were inspired by privacy issues, such as safe searches over encrypted cloud data. However, in practice, most cloud servers do not represent a single owner; rather, they allow multiple owners to share the benefits provided by cloud servers. We systematically design a novel secure search protocol to enable cloud servers to perform secure search without knowing the actual data of both keywords and trapdoors. We propose a novel Additive Order and Privacy Preserving Function family to rank search results while maintaining the privacy of relevance scores between keywords and files.

Cui, Y., et al. [2020] proposes in encrypted keyword index generation, an ABMKS with only multiplication operations is used. Searchable encryption is a critical tool for providing safe search over encrypted cloud data while maintaining data confidentiality and privacy. Consider a protected search service that provides fine-grained and search features, known as attribute-based multiple keyword search (ABMKS), which can be viewed as an extension of searchable encryption. Finally, our scheme is both effective and stable, as shown by the security and performance analyses.

Krishna, C. R., & Handa, R. [2016] proposed a search scheme meets all of the security standards suggested in current methods in the literature, but it also effectively offers searching and dynamic updates performance. Earlier search schemes used to create a static index that only supported searching. To handle dynamic updates as well as searching, the created index is made dynamic rather than static. The experimental results show that the proposed dynamic search scheme is successful because it efficiently retrieves documents with modified versions.

Sun, J., et al. [2019] by integrating the ciphertext policy attribute-based encryption (CP-ABE) and auditing concepts, this paper proposes a practical multi-keyword searchable encryption scheme for data integrity verification and attribute revocation. Although traditional searchable encryption technology can provide data security and retrieval capabilities, there are a few major issues to consider. Focusing on these issues, the scheme on the one hand supports multi-keyword search, which reduces the number of irrelevant documents returned by the cloud server by narrowing the search scope, and on the other hand can enforce successful attribute revocation by entrusting ciphertext updates to the powerful cloud server, preventing access by unauthorized users.

In contrast to the majority of these works, there is a multi-word search method that secures an ECC algorithm in order to resist exposure range and correlation attacks [12]. Eventually, the proposed scheme is compared to the DH algorithm, which focuses on Multi keyword Ranked Searchable Encryption schemes, in order to test its effectiveness.

3. THE PROPLEM DEFINITION

Cloud storage provides users with various advantages such as lower deployment and maintenance costs, scalability, increased performance, system and location independence, and so on. Using cloud storage to store sensitive data raises the risk of information leakage by a factor of ten. Adversaries (malicious insiders and outsiders) constantly track the outsourced data for useful insights. As a result, end users who use these third-party services see data protection (including both storage and computation security) as the key obstacle to cloud adoption because data is continuously vulnerable to security threats during its life cycle, i.e., at various stages such as data generation and collection, data transfer, data storage, data exchange, data application, and destruction.

CSP provides such security mechanisms in order to provide data security (e.g. data encryption). However, end-users do not trust these solutions since the entire system for providing protection (i.e., hidden keys used for encryption) is under the control of CSP and this information can be exposed at any time by malicious insiders for financial gain. As a result, it is desirable that data be safe both at rest and in transit. One of the steps to ensure this is to transfer ownership of the cryptographic primitives. End users are allowed to create and maintain the secret keys used for encryption in order to accomplish this. Thus, data is encrypted by end users until it leaves the users' borders, providing protection both in transit and storage.

While encryption ensures data protection, it makes routine search operations difficult. This is due to the fact that plain-text queries cannot be used explicitly because they expose the search terms to CSP. Furthermore, due to the lack of a search protocol that works for both plain-text queries and encrypted documents, the encrypted documents referring to these plain-text queries cannot be retrieved. As a result, searchable encryption schemes emerged, which produce encrypted search queries and provide users with search capabilities.

It is necessary to investigate an effective SE scheme in order to reduce the financial burden associated with cloud use. Various schemes have been suggested by the researchers to aid in efficient retrieval. These schemes were able to speed up the search process, but they had limited search capability, as they only supported conjunctive searching. Other Researchers, on the other hand, worked to provide improved search capabilities but lacked the desired degree of search performance, resulting in a tradeoff between search efficiency and search capability.

4. THE PROPOSED SCHEME

The proposed scheme first conducts a conjunctive keyword search, in which documents containing the entire query words are easily retrieved and sorted based on their relevance to the query. If the number of obtained results exceeds the number of desired results, the search is efficient. If, on the other hand, the results fall below the end user-specified threshold, disjunctive searching is used. The records containing the subset of query words are included in the results of disjunctive searching. The obtained results are rated and shared with the end users. As a result, the proposed hybrid search based on the forward and inverted indexes strikes a balance between search efficiency and capability.

The proposed scheme should meet the following design objectives..

1. Multi-keyword search: It can handle several keywords in the search query, i.e., it should mimic real user search activity by allowing a different number of search terms.
2. Ranked retrieval: The answer to the search query should be ordered based on its relevance to the query, reducing post-processing computational overheads for end users.

Multi-keyword searching represents end-user search activity. As a result, an encrypted search query containing all of the search terms entered by the end user is created in this process. The method used to generate the search query requires less computing resources and can be easily created by end users using any computer.

It consists of comparing the question to the outsourced indexes. In the literature, various symmetric search schemes have been proposed. Methods such as document clustering and keyword binning based on different indexes are proposed in this phase to reduce the amount of time needed for document retrieval.

4.1. The Proposed Search Process

The proposed scheme's search protocol is a recursive procedure on the tree known as the "Iterative Deepening Depth-First Search" (ID-DFS) algorithm. IDDFS combines the space-efficiency of depth-first search with the speed of breadth-first search. IDDFS calls DFS for various depths starting from a given value. DFS is prohibited from going beyond the depth specified in each call. [17]. the algorithm's basic principle is to begin with a start node and then examine the node's first child. It then looks at that node's first child (grandchild of the starting node), and so on, until a node has no more children. It then advances one step and examines the next boy. If there are no more children, it advances one step higher, and so on until it finds more children or reaches the start node. If the goal node has not been reached after returning from the last child of the start node, the goal node cannot be found because all nodes have been traversed by that stage.

IDDFS combines the space-efficiency of depth-first search with the completeness of breadth-first search (when the branching factor is finite). When the path cost is a non-decreasing function of the node depth, it is optimal.

Iterative deepening may seem inefficient because it visits states multiple times, but it turns out to be not so expensive because most nodes in a tree are in the bottom level, so it does not matter if the upper levels are visited multiple times.

Implementation:

Step 1: -For each child of the current node
Step 2: - If it is the target node, return
Step 3: - If the current maximum depth is reached, return.
Step 4: - Set the current node to this node and go back to 1.
Step 5: - After having gone through all children, go to the next child of the parent (the next sibling)
Step 6: - After having gone through all children of the start node, increase the maximum depth and go back to 1.
Step 7: - If we have reached all leaf (bottom) nodes, the goal node doesn't exist.

4.2. Retrieval Process of Proposed Scheme

The process of finding data (generally documents) in the form of text that matches the information required from a set of documents stored on a computer is known as information retrieval (IR). Incorrect user requests are a common problem on IRs; this is caused by user weaknesses in representing their needs in the query. Researchers have suggested numerous solutions to address these limitations; in this review, we proposed an approach based on the FP-Growth algorithm for the quest for frequent itemsets. The key to storing keywords is the cloud storage server, which is denoted by the letter K. ($keyword_0$, $keyword_1$, $keyword_2$, and $keyword_3$). Calculate the support for a single itemset by traversing each record and seeing if it includes an itemset.

Procedure: FPgrowth (DB, ξ)

Step 1: - Define and clear F-List : F[];

Step 2: - foreach T transaction Ti in DB do
Step 3: - Foreach Item aj in Ti do
Step 4: - F[ai] ++;
Step 5: - end
Step 6: - end
Step 7: - Sort F[];
Step 8: - Define and clear the root of FP-tree : r;
Step 9: - foreach T transaction Ti in DB do
Step 10: - Make Ti ordered according to F;
Step 11: - Call Construct T tree(Ti, r);
Step 12: - end
Step 13: - foreach item a_i in I do
Step 12: - Call Growth(r, a_i, ξ);
Step 13: - end

This algorithm begins by compressing the input database, resulting in a frequent pattern tree case. The compressed database is then divided into a few conditional databases, each representing a single specific frequent pattern. Finally, mining of each database is done separately. As a result, the search costs are greatly reduced, resulting in strong selectivity.

5. PERFORMANCE EVALUATION

Cloud computing has made use of a variety of data mining techniques [12]. The Apriori algorithm is well-known, but the current implementation relied on the original Apriori for the cloud computing paradigm. Some have attempted parallelism, but have been unable to reduce the number of steps in the Apriori algorithm. To determine the performance of the comparative algorithms, the parameters for which the algorithms must be evaluated must be specified. The output of the Apriori algorithm and the FP-growth algorithm for various support levels is discussed in this section. It demonstrates that the FP-growth algorithm takes significantly less time to execute than the Apriori algorithm for any Support stage. With the lowest support standard, the execution time is reduced.

As a result of the experimental analysis, it is clear that the FP-growth algorithm outperforms the Apriori algorithm.

i) Search Accuracy

It is desirable for a SE scheme to be able to provide high search accuracy as well as high search efficiency in order for it to be basically adopted. In comparison to the previous scheme, which involved scanning the entire document set, the proposed search scheme reduces the total search time required to find the relevant documents. As a result, the proposed scheme improves search performance.

Table 5.1: - Comparison of Accuracy

No. of Documents	Existing Approach	Proposed Approach
5	90	98
10	83	95
15	77	91
20	71	87
25	65	84

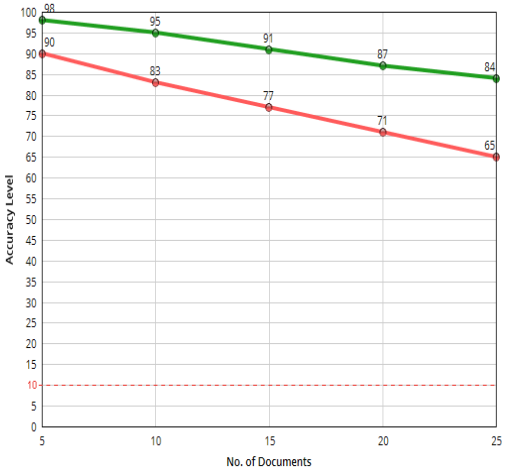

Fig.5.1: - Search Accuracy Level

ii) Top k-files results

Any file from the cloud can be downloaded by registered users. They can filter the files based on a keyword. Users will be shown the files associated with that keyword. Users may also access the files based on their highest rank. Each download of a file increases the rank of that file. As a result, users can easily obtain information about the top rated data, as shown in fig.5.2.

Table 5.2: - Comparison of top K-results

No. of files	Existing Approach (Apriori)	Proposed Approach (FPgrowth)
200	0.6	0.3
400	0.8	0.4
600	1.0	0.5
800	1.4	0.8
1000	1.8	0.8

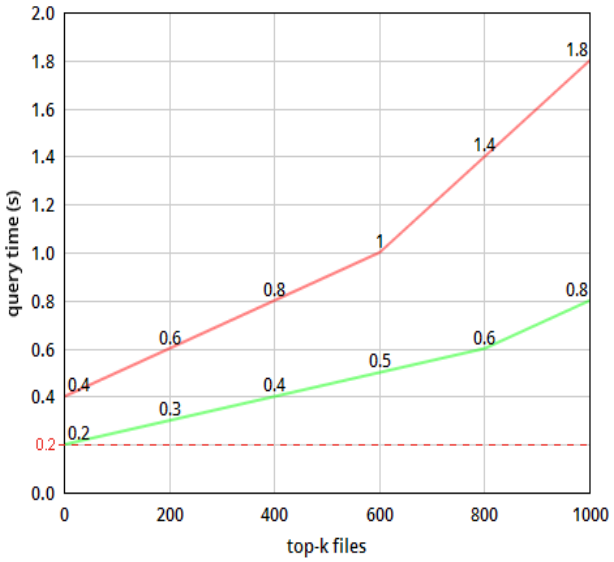

Fig 5.2: - Top K-results

5. CONCLUSION

This paper proposes a reliable, effective, and dynamic search scheme that not only supports accurate multi-keyword ranked search but also dynamic deletion and insertion of documents. The presented performance characteristics demonstrate the various points. The results of the experiment show that, to some degree, our scheme can significantly increase the search efficiency of privacy-preserving multi-keywords ranked search while also guaranteeing the accuracy of the search results. The integrity audit for data that is outsourced to the cloud has been thoroughly examined.

6. REFERENCES

[1] Sun, J., Hu, S., Nie, X., & Walker, J. (2019). Efficient ranked multi-keyword retrieval with privacy protection for multiple data owners in cloud computing. IEEE Systems Journal, 14(2), 1728-1739.

[2] Zhang, W., Xiao, S., Lin, Y., Zhou, T., & Zhou, S. (2014, June). Secure ranked multi-keyword search for multiple data owners in cloud computing. In 2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (pp. 276-286). IEEE.

[3] Cui, Y., Gao, F., Shi, Y., Yin, W., Panaousis, E., & Liang, K. (2020). An efficient attribute-based multi-keyword search scheme in encrypted keyword generation. IEEE Access, 8, 99024-99036.

[4] Krishna, C. R., & Handa, R. (2016, January). Dynamic cluster based privacy-preserving multi-keyword search over encrypted cloud data. In 2016 6th International Conference-Cloud System and Big Data Engineering (Confluence) (pp. 146-151). IEEE.

[5] Sun, J., Ren, L., Wang, S., & Yao, X. (2019). Multi-keyword searchable and data verifiable attribute-based encryption scheme for cloud storage. IEEE Access, 7, 66655-66667.

[6] Örencik, C., & Savaş, E. (2012, March). Efficient and secure ranked multi-keyword search on encrypted cloud data. In Proceedings of the 2012 Joint EDBT/ICDT Workshops (pp. 186-195).

[7] Chuah, M., & Hu, W. (2011, June). Privacy-aware bedtree based solution for fuzzy multi-keyword search over encrypted data. In 2011 31st International Conference on Distributed Computing Systems Workshops (pp. 273-281). IEEE.

[8] Miao, Y., Ma, J., Liu, X., Wei, F., Liu, Z., & Wang, X. A. (2016). m 2-ABKS: Attribute-based multi-keyword search over encrypted personal health records in multi-owner setting. Journal of medical systems, 40(11), 1-12.

[9] Rane, D. D., & Ghorpade, V. R. (2015, January). Multi-user multi-keyword privacy preserving ranked based search over encrypted cloud data. In 2015 International Conference on Pervasive Computing (ICPC) (pp. 1-4). IEEE.

[10] Zarezadeh, M., Mala, H., & Ashouri-Talouki, M. (2020). Multi-keyword ranked searchable encryption scheme with access control for cloud storage. Peer-to-Peer Networking and Applications, 13(1), 207-218.

[11] Peng, T., Lin, Y., Yao, X., & Zhang, W. (2018). An efficient ranked multi-keyword search for multiple data owners over encrypted cloud data. IEEE Access, 6, 21924-21933.

[12] Wodi, B. H., Leung, C. K., Cuzzocrea, A., & Sourav, S. (2019, December). Fast privacy-preserving keyword search on encrypted outsourced data. In 2019 IEEE International Conference on Big Data (Big Data) (pp. 1-10). IEEE.

[13] Mayan, J. A., Arifa, S., & Pavithra, R. (2016, December). Semantic based multi lexical ranking technique for an effective search in protected cloud. In 2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT) (pp. 570-576). IEEE.

[14] Pandiaraja, P., & Vijayakumar, P. (2017, February). Efficient multi-keyword search over encrypted data in untrusted cloud environment. In 2017 Second International Conference on Recent Trends and Challenges in Computational Models (ICRTCCM) (pp. 251-256). IEEE.

[15] Dai, H., Ji, Y., Yang, G., Huang, H., & Yi, X. (2019). A privacy-preserving multi-keyword ranked search over encrypted data in hybrid clouds. IEEE Access, 8, 4895-4907.

[16] Ananthi, S., Sendil, M. S., & Karthik, S. (2011, July). Privacy preserving keyword search over encrypted cloud data. In International Conference on Advances in Computing and Communications (pp. 480-487). Springer, Berlin, Heidelberg.

[17] Handa, R., Rama Krishna, C., & Aggarwal, N. (2020). Efficient privacy-preserving scheme supporting disjunctive multi-keyword search with ranking. Concurrency and Computation: Practice and Experience, 32(2), e5450.

[18] Zittrower, S., & Zou, C. C. (2012, December). Encrypted phrase searching in the cloud. In 2012 IEEE Global Communications Conference (GLOBECOM) (pp. 764-770). IEEE.

[19] Shen, Z., Shu, J., & Xue, W. (2016). Keyword search with access control over encrypted cloud data. IEEE Sensors journal, 17(3), 858-868.

[20] Gampala, V., & Malempati, S. (2016). An efficient Multi-Keyword Synonym Ranked Query over Encrypted Cloud Data using BMS Tree. International Journal of Applied Engineering Research, 11(1), 738-743.