NOVEL APPROACH FOR KEYWORD SEARCH OVER SECURE SENSITIVE DATA ON CLOUD ENVIRONMENT UNDER SECTION 43A OF THE IT ACT USING FUZZY MULTI KEYWORD SEARCH APPROACH

C.M. Varun[1], R.P. Anto Kumar[2]

[1]Assistant Professor, Department of Computer Science and Business Systems, R.M.K. Engineering College, Kavaraipettai, Tamil Nadu
[2]Professor, Department of Computer Science and Engineering, St. Xavier's Catholic College of Engineering, Chunkankadai, Tamil Nadu

cmvarun87@gmail.com[1]
antokumar@sxcce.edu.in[2]

Abstract: Cloud servers have become a preferred platform for individuals and organizations to outsource and manage data. However, protecting sensitive information through encryption creates challenges for effective data retrieval. Traditional search methods mainly rely on exact keyword matching, which limits usability when dealing with synonyms or misspellings. To address this issue, this study proposes a novel fuzzy keyword ranked search framework for encrypted cloud data. The approach integrates a fuzzy logic wordbook to expand search terms, include synonyms, and handle spelling variations. In the data owner phase, documents are encrypted using the ERA algorithm, while index tables are generated through preprocessing, clustering (IFFC), summarization (LSA), and score calculation (ASC). Both the documents and the index tables are further protected using ERA encryption and freshness is maintained with the SHA-516 algorithm. During the DU phase, the WFKS method facilitates secure and ranked document retrieval from the CSP. Experimental evaluation demonstrates that the proposed method achieves faster, more accurate, and more secure retrieval compared with existing techniques, thereby improving efficiency and privacy in encrypted cloud environments.

Key words: Encoded RSA and AES (ERA), Enhanced Farthest First Clustering (EFFC) algorithm, Adaptive Scoring Mechanism (ASM), Wordbook-based Fuzzy Keyword Search (WFKS), Data Owner (DO), Data User (DU), Cloud Service Provider (CSP), Secure Hash Algorithm-516 (SHA-516), and Dynamic Data Management.

1. INTRODUCTION

Cloud computing is a rapidly growing model of distributed computing and storage, offering users quick, convenient services and data backups through public, private, and hybrid deployments [1–4]. By reducing local storage needs and capital expenditure on hardware and maintenance, it enables cost-effective scalability. However, massive amounts of sensitive data—such as medical files and financial statements—are now stored on cloud servers [5–6], raising privacy concerns. To protect confidentiality, data must be encrypted before upload [7].

Encryption secures data but complicates keyword searching, since cloud servers cannot read encrypted content. Downloading and decrypting all data is impractical. Searchable Encryption (SE) addresses this by allowing keyword searches directly over encrypted datasets [8–10]. SE prevents clients from downloading or decrypting entire files, instead using encrypted "trapdoors" for secure retrieval [11–13]. Still, some risk of information leakage remains through search patterns and indexing [14].

In a typical SE setup, the data owner encrypts documents and keywords before uploading them. When a data user submits a keyword token, the server applies a secure search algorithm and returns matching encrypted documents [15–17]. Current research focuses on improving performance, expressiveness, and usability [18–19].To enhance search accuracy and efficiency, fuzzy keyword ranked search techniques have been proposed. These account for synonyms and spelling variations, ranking results by relevance while maintaining encryption.This paper explores a novel keyword search approach over encrypted data. Section 2 reviews existing work, Section 3 introduces our method, Section 4 evaluates results, and Section 5 concludes.

2. RELATED WORK

Guo et al. [20] designed a secure multi-keyword ranked search technique for encrypted cloud data by integrating the vector space model with TF–IDF weighting and cosine similarity to measure query–document relevance. To speed up retrieval, they implemented an index tree built upon Bloom filters for efficient document matching. Yunyun Wu et al. [21] introduced PEMKS, a system capable of handling n-keyword searches and negation operations. Although effective, its dependence on the Diffie–Hellman key exchange protocol increases computational overhead. Hui Yin et al. [22] developed a secure multi-keyword ranked search architecture for multiple data owners that uses random temporary keys to generate secure indices and enables authorized users to issue secure query keys without exposing the underlying index keys. Zhong et al. [23] proposed a dynamic multi-keyword fuzzy search model leveraging Locality Sensitive Hashing (LSH) and Bloom filters to create both index and query vectors. Their framework incorporates a balanced binary tree with a Top-k search mechanism but encounters scalability and computational challenges with KNN during file selection. Jianfei Sun et al. [24] offered a ranked multi-keyword search approach that safeguards the anonymity of multiple data owners and enables the cloud to securely execute Top-k searches while maintaining keyword and trapdoor privacy. Maryam Hozhabr et al. [25] presented a secure multi-keyword ranked search system that accommodates dynamic document operations and fine-grained access control. By employing B+ tree indexing for sublinear search time and MB-tree structures for enhanced security, their model effectively restricts unauthorized access in “Honest-but-Curious” cloud environments.

3. NOVEL METHODOLOGY FOR RANKED FUZZY KEYWORD SEARCH ON ENCRYPTED CLOUD DATA

In recent years, cloud computing has become popular for managing personal data due to its cost-effectiveness and flexibility. However, transferring data to the cloud reduces user control and exposes it to potential cyber threats despite CSP security measures. To address this, stronger safeguards are needed for sensitive data. This study introduces novel fuzzy keywords to rank searches over encrypted cloud data, using wordbooks to handle synonyms and misspellings via fuzzy logic.

3.1 Data Owner

In this stage, the Data Owner (DO) enhances security for the Cloud Service Provider (CSP) by outsourcing encrypted documents along with their associated keywords. Using the ERA (Encoded RSA and AES) approach, the documents are encrypted before transmission. The DO then constructs an index table, which is developed through four key phases: clustering, preprocessing, keyword summarization, and score calculation. To maintain document integrity and ensure its timeliness, the Data User applies the SHA-512 algorithm for freshness verification.

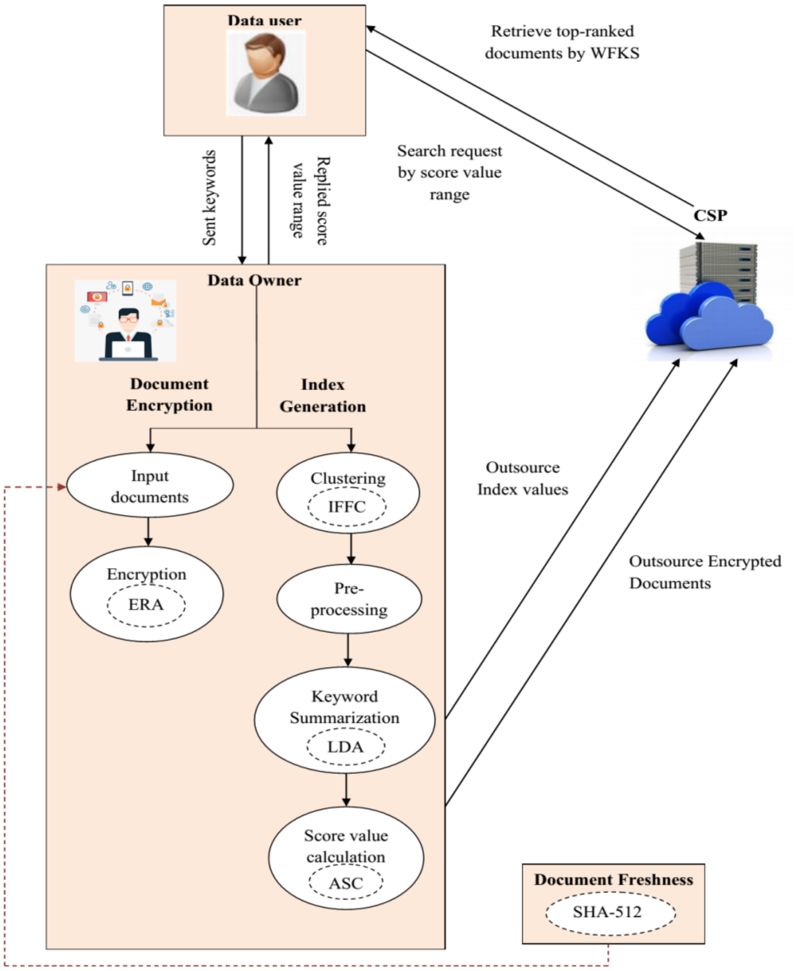


Figure 1: Block diagram for the proposed methodology

3.1.1 Document Encryption

Initially, the ERA technique encrypts the documents received from the Data Owner (DO). The Advanced Encryption Standard (AES) works with a fixed block size of 128 bits (16 bytes), arranging the data in 4×4 byte matrices. The number of encryption rounds depends on the key length. To strengthen confidentiality, the ERA framework integrates RSA with AES: the data owner first generates RSA key pairs, encrypts the documents using AES, and then secures the AES key itself with RSA. An additional XOR encoding layer is applied to safeguard keys during transmission. In this process, RSA's public key is used to encrypt the AES key prior to document encryption, while the private key is retained for secure decryption.

$$E_c = P^l \bmod n \oplus N(q_i) \tag{1}$$

Here, C represents the cipher value produced during document encryption, K denotes the AES encryption key, and F indicates the ASCII-encoded version of the input document's file name. The AES encryption process consists of four main steps:a) Byte substitutionb) Row shiftingc) Column mixingd) Adding the round key

AES applies four steps per round: Substitute Bytes replaces each byte via a nonlinear lookup table; Shift Rows cyclically shifts rows to the left; Mix Columns multiplies each column polynomial by a fixed polynomial; and Add Round Key XORs the state with round keys derived from the encryption key.

3.1.2 Index generation

After encrypting the documents, the Data Owner (DO) generates an index table. This process involves several stages: clustering the data, carrying out pre-processing, summarizing keywords, and finally calculating the score values.

3.1.2.1 Clustering

The Improved Farthest First Clustering (IFFC) method groups documents to improve retrieval accuracy. Unlike K-Means, it selects cluster centers by choosing the point farthest from existing centers, reducing reallocation and speeding up clustering. Initial centroids are chosen using medoid values, and taxicab distance refines distance computation for better grouping. The $Q_s = \{q_1, q_2, q_3, ..., q_n\}, or\, i, \forall i = 1,2,3,....,n$ DO documents are first obtained, and the centroid value is determined using the medoid computation, which is represented as follows:

$$O_i = \frac{N(q_i)}{2} \tag{2}$$

Here, O_i represents the medoid value and denotes the total number of documents. The process begins by identifying the data point farthest from the initial centroid. Next, the point most distant from both previously chosen locations is selected. Finally, the data is clustered using a method based on calculating the Manhattan (taxicab) distance.

$$Z_d = \sum_{i=1}^{n} |O_i - q_i| \tag{3}$$

Once the distances between documents are calculated, each document is assigned to its closest centroid. The centroids of the newly formed clusters are then recalculated. This iterative process of centroid adjustment and distance computation continues until all documents are appropriately grouped into clusters. The final clustered output is then represented as follows:

$$H_l = \{h_1, h_2, ..., h_n\} or\ h_i, i = 1,2,.....,n \tag{4}$$

Where h_n specify the clusters, H_l as well as the cluster set respectively

3.1.2.2 Pre-processing

After clustering the documents, pre-processing is performed. Sentences are segmented, and common stop words such as “the” or “is” are removed. Variations in spelling are also generated to ensure broader document coverage. Stemming is then applied to extract the root form of words—for instance, “connect” serves as the base for “connected,” “connecting,” and “connections.” This process trims unnecessary suffixes, reduces the total number of terms, improves matching accuracy, and optimizes time and memory usage. The outcome of the pre-processing stage is presented as follows:

$$G_l = g_i, = 1,2,...,n \tag{5}$$

Where, G_l the pre-processing output set and g_i specification all the documents that were pre-processed.

3.1.2.2 Summarization

Latent Semantic Analysis (LSA) summarizes cluster terms by identifying semantic relationships between words and documents. It creates a document-term matrix, applies Singular Value Decomposition (SVD), and selects key sentences for summarization. Rows represent words, columns represent documents, enabling extraction of the most meaningful phrases.

(a) Input matrix

The matrix $S_l = [s_1, s_2, ..., s_n]$ organizes terms across sentences, with each column serving as a vector that captures the weighted frequency of terms appearing in the provided document. This structure is built for a document containing a specific count of phrases and sentences. It tends to be sparse because individual words rarely appear in all phrases.

b) Matrix Factorization Using Singular Value Decomposition

To analyze the relationship between words and phrases, mathematicians often turn to singular value decomposition. This technique involves breaking down the input matrix into three separate matrices:

$$S = R \sum P^T \tag{6}$$

Where, $\sum$ denotes diagonal Eigen values in decreasing order, P^T symbolizes the transposition of an orthogonal matrix, and S indicates the input matrix of sentence × words. It also represents a matrix that describes the original rows of the input matrix as a vector of extracted ideas.

c) Generation of Summary

The process generates a summary by selecting key sentences from SVD results using a cross technique. Sentence vectors' lengths determine importance, based on user-specified concepts. The longest vectors form the summary, and the clustering's main keyword is identified from recurring terms across documents.

3.1.2.3Computation of Scoring Metrics

In this method, the Adaptive Score Calculation (ASC) technique is applied to evaluate scores for the key terms extracted in the summary. ASC integrates several approaches, including Information Gain (IG), Jaccard similarity, Dice coefficient, and Chi-square analysis.

For IG: This metric employs entropy to evaluate the influence of data alterations on the dataset's overall purity. The computation of IG is detailed below:

$$I_g = E_{bs} - E_{as} \tag{7}$$

Where, I_g information gained, E_{bs} and E_{as} definitions are given before and after the keyword is divided.

Jaccard: It measures document similarity on a scale from 0 to 1, where 0 means completely different and 1 means identical. Values in between indicate the degree or likelihood of similarity between the two texts. The following is how the jaccard computation is expressed:

$$J_d = \frac{|k_i \cap k_{i+1}|}{|k_i \cup k_{i+1}|} \tag{8}$$

where the document's keyword is specified k_i and k_{i+1} along J_d with the jaccard value.

Dice: The Dice coefficient, commonly called the overlap index, acts as a statistical tool for evaluating the similarity between two sets of keywords. Below is the formula for computing the Dice similarity coefficient:

$$D_c = 2 * \frac{|k_i \cap k_{i+1}|}{(|k_i| + |k_{i+1}|)} \tag{9}$$

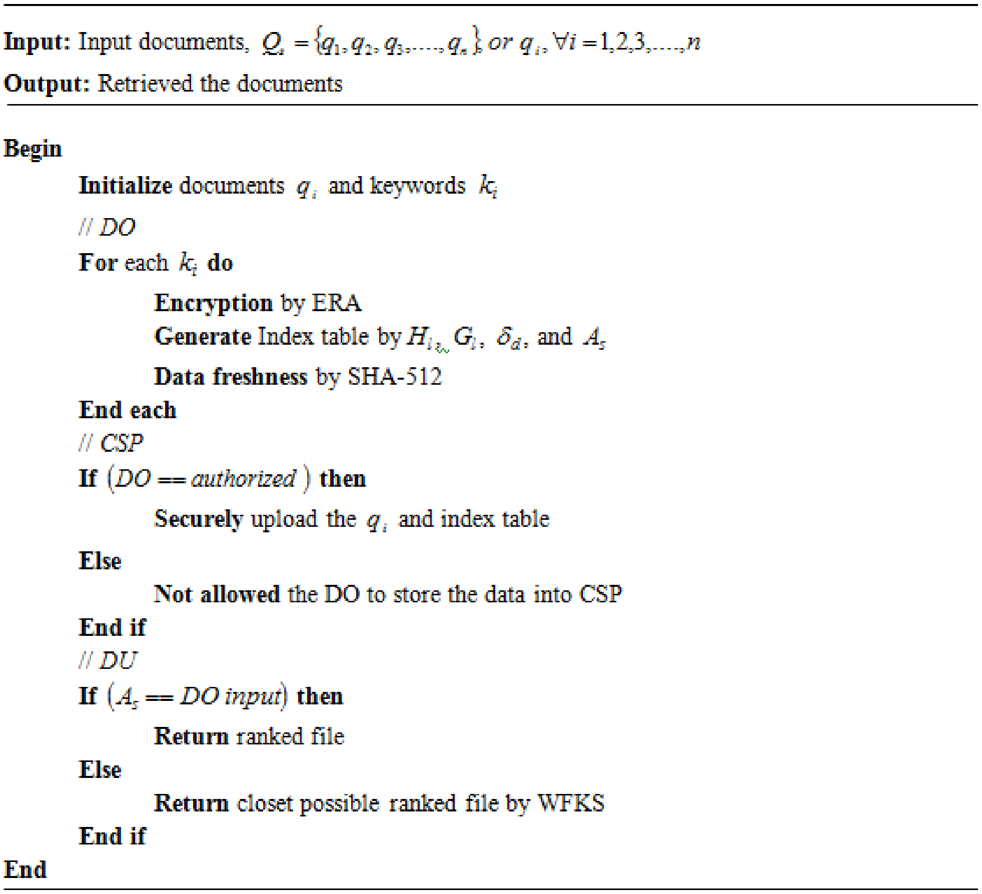
Chi: This metric evaluates the difference between two terms. The chi-square computation is outlined below:

$$C_s = \sum \frac{(v_d - e_d)^2}{e_d} \tag{10}$$

Where, e_d denotes the predicted value, v_d indicates the actual value, and C_s defines the chi square output. The following symbols represent the calculation's A_s adaptive process:

$$A_s = I_g + J_d + D_c + C_s \tag{11}$$

The score value range of the cluster is also provided in the index after the score value calculation. Here, the same ERA technique is also used to encrypt the keywords.The DO ensures document freshness using SHA-512, updating only changed parts instead of full re-encryption. This 512-bit hash works through padding, length appending, buffers, 1024-bit block processing, and 80 rounds to produce a message digest.The CSP securely stores encrypted documents and keywords for upload and access.Authorized users query via score ranges; WKFS supports fuzzy search with synonyms. Retrieved files are decrypted using RSA and AES keys.The pseudocode for the proposed approach is illustrated in Figure 2. It comprehensively outlines the DO, DU, and CSP procedures.



```
Input: Input documents, $Q_s = \{q_1, q_2, q_3, \ldots, q_n\}$ or $q_i, \forall i = 1,2,3,\ldots,n$
Output: Retrieved the documents

Begin
    Initialize documents $q_i$ and keywords $k_i$
    // DO
    For each $k_i$ do
        Encryption by ERA
        Generate Index table by $H_i$, $G_i$, $\delta_d$, and $A_s$
        Data freshness by SHA-512
    End each
    // CSP
    If $(DO == authorized)$ then
        Securely upload the $q_i$ and index table
    Else
        Not allowed the DO to store the data into CSP
    End if
    // DU
    If $(A_s == DO\ input)$ then
        Return ranked file
    Else
        Return closet possible ranked file by WFKS
    End if
End
```

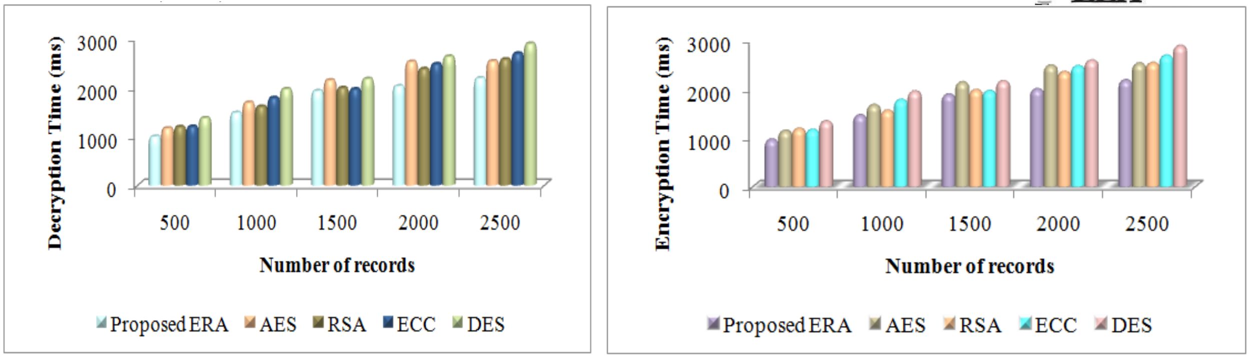
Figure 2: Pseudo code for the proposed methodology

4. RESULT AND DISCUSSION

This section evaluates the efficiency of the proposed keyword search over encrypted cloud data, implemented in Java. The performance study uses Electronic Medical Records (EMRs), commonly outsourced by hospitals to remote servers for easier storage and maintenance. Each record includes predefined patient attributes such as age, gender, and illness.

4.1Evaluation of Encryption Efficiency for Documents and Index Tables

This analysis evaluates the proposed ERA-based encryption technique for documents and index tables by examining encryption duration, decryption speed, and overall security strength. It contrasts this approach with established methods such as AES, RSA (Rivest-Shamir-Adleman), ECC (Elliptic Curve Cryptography), and DES (Data Encryption Standard).



(a)(b)

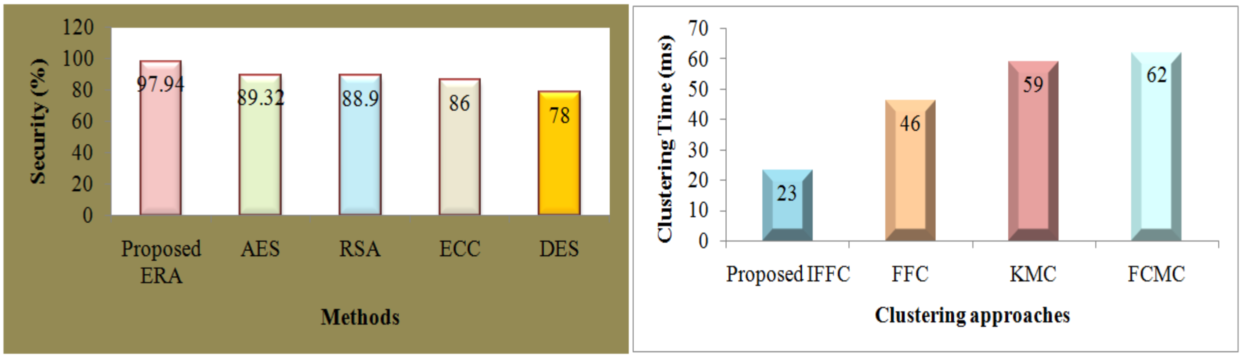
Figure 3: Graphical plot for (a) Encryption time analysis and (b) Decryption time analysis

The security is next examined in light of the suggested and current research approaches. The following is the mathematical expression for the security analysis:

$$S_l = \frac{A_D}{T_D} \times 100 \qquad (12)$$

Where, T_D determines the total quantity of data, A_D denotes the attacked data, and S_l defines the security level.

Security level refers to the extent to which the technique blocks unauthorized individuals from accessing the content. In this evaluation, the innovative ERA approach achieves a 97.94% security rating, surpassing the 89.32% for AES, 88.9% for RSA, 86% for ECC, and 78% for DES among existing methods. Consequently, the performance assessment highlights that this proposed solution delivers superior protection relative to contemporary techniques. The graphical representation of these security outcomes appears in Figure 4(a).



(a)(b)

Figure 4: Graphical plot for the (a)security analysis, (b) clustering time analysis

4.2 Evaluation of Cluster Grouping Effectiveness

The proposed IFFC-based clustering is evaluated against FFC, KMC, and FCMC for time and accuracy. Figure 4(b) shows IFFC clustering in just 23 s, outperforming FFC (28 s), KMC (59 s), and FCMC (62 s), clearly demonstrating its superior efficiency.

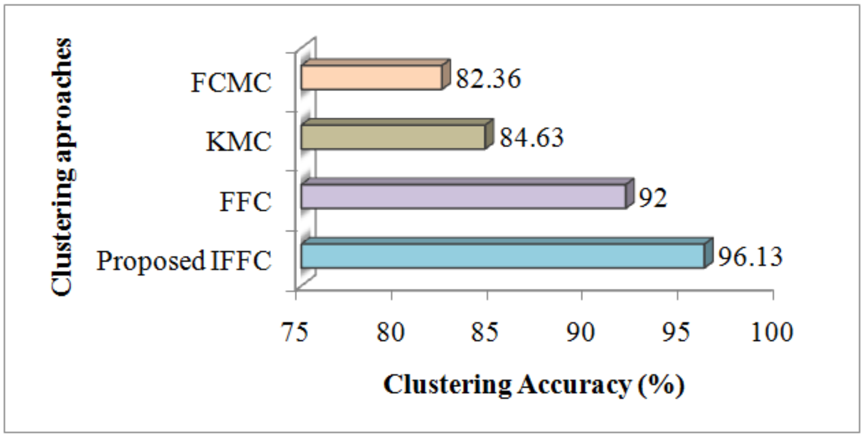
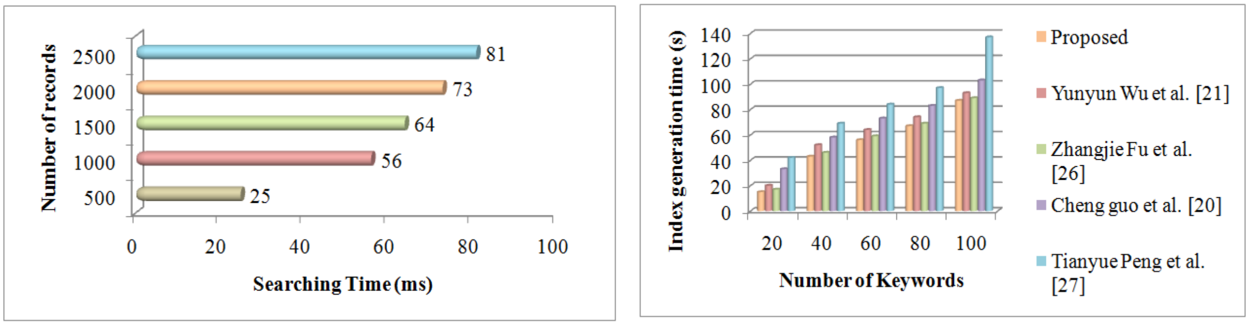


Figure5: Illustration of the performance evaluation for the proposed IFFC approach compared to established algorithms, assessed via the clustering accuracy measure.

Figure 6 displays the clustering accuracy performance study of the proposed IFFC in comparison to the existing methods. The clustering accuracy of the proposed methodology is 96.13%, while the current methodologies' respective clustering accuracy is 92%, 84.63%, and 82.36%. Consequently, it is evident that the IFFC-based clustering method yields better outcomes than earlier research.

4.3 Evaluation of Index Creation and Query Execution Efficiency

This section employs visual graphs to evaluate the durations for constructing indexes and executing searches within the proposed techniques.



(a)(b)

Figure6:(a) Evaluation of the response time required for information retrieval using the proposed technique(b) Assessment of the time involved in constructing the index table.

Figure 6(a) illustrates the proposed method’s search time across varying record sizes. It achieves 25 ms for 500 records and scales to 56, 64, 73, and 81 ms for 1,000–2,500 records, showing a gradual increase with larger datasets. Figure 6(b) highlights its faster index table creation compared with existing methods. For 20–100 keywords, it completes in 15–87 ms, outperforming Tianyue Peng et al. [27] and other approaches, demonstrating higher efficiency in both search and index generation.

5. CONCLUSION

Cloud computing enables organizations to access networks, computing power, and storage remotely, but safeguarding sensitive data requires robust encryption. This study proposed an enhanced approach for keyword search over encrypted data involving DO, DU, and CSP processes. Using Electronic Medical Records (EMR) as the test case, the method was evaluated against RSA, ECC, AES, and DES in terms of encryption/decryption time, security, clustering efficiency, and index generation. Results show that the proposed ERA methodology achieved faster encryption and decryption, improved clustering accuracy, and stronger security (97.94%) compared with existing techniques. Index creation and search times were also optimized. Overall, the findings demonstrate the approach's effectiveness for secure keyword searching in cloud environments, with potential for future improvements through broader dictionaries and advanced methods.

REFERENCES

1. P. Shanthi, and A. Umamakeswari, "Privacy preserving time efficient access control aware keyword search over encrypted data on cloud storage", Wireless Personal Communications, vol. 109, no. 4, pp. 2133-2145, 2019.
2. Hua Dai, Yan Ji, Geng Yang, Haiping Huang, and Xun Yi, "A privacy-preserving multi-keyword ranked search over encrypted data in hybrid clouds", IEEE Access, vol. 8, pp. 4895-4907, 2019.
3. Hoang Pham, Jason Woodworth, and Mohsen Amini Salehi, "Survey on secure search over encrypted data on the cloud", Concurrency and Computation: Practice and Experience, vol. 31, no. 17, pp. e5284, 2019.
4. Jassim R. Mlgheit, Essam H. Houssein, and Hala H. Zayed, "Efficient Privacy Preserving of Multi-keyword Ranked Search Model over Encrypted Cloud Computing", In 2018 1st International Conference on Computer Applications & Information Security (ICCAIS), IEEE, pp. 1-6, 2018.
5. Alfredo Cuzzocrea, Carson K. Leung, Bryan H. Wodi, S. Sourav, and Edoardo Fadda, "An effective and efficient technique for supporting privacy-preserving keyword-based search over encrypted data in clouds", Procedia Computer Science, vol. 177, pp. 509-515, 2020.
6. Yanrong Liang, Yanping Li, Qiang Cao, and Fang Ren, "VPAMS: Verifiable and practical attribute-based multi-keyword search over encrypted cloud data", Journal of Systems Architecture, vol. 108, pp. 101741, 2020.
7. Debasis Das, and Sumit Kalra, "An Efficient LSI Based Multi-keyword Ranked Search Algorithm on Encrypted Data in Cloud Environment", In 2020 International Wireless Communications and Mobile Computing (IWCMC), IEEE, pp. 1777-1782, 2020.
8. Zhangjie Fu, Lili Xia, Xingming Sun, Alex X. Liu, and Guowu Xie, "Semantic-aware searching over encrypted data for cloud computing", IEEE Transactions on Information Forensics and Security, vol. 13, no. 9, pp. 2359-2371, 2018.
9. Yinbin Miao, Jianfeng Ma, Ximeng Liu, Zhiquan Liu, Limin Shen, and Fushan Wei, "VMKDO: Verifiable multi-keyword search over encrypted cloud data for dynamic data-owner", Peer-to-peer Networking and Applications, vol. 11, no. 2, pp. 287-297, 2018.
10. Lili Zhang, Yuqing Zhang, and Hua Ma, "Privacy-preserving and dynamic multi-attribute conjunctive keyword search over encrypted cloud data", IEEE Access, vol. 6, pp. 34214-34225, 2018.

11. Bo Lang, Jinmiao Wang, Ming Li, and Yanxi Liu, "Semantic-based compound keyword search over encrypted cloud data", IEEE Transactions on Services Computing, 2018.
12. DVN Siva Kumar, and P. Santhi Thilagam, "Approaches and challenges of privacy preserving search over encrypted data", Information Systems, vol. 81, pp. 63-81, 2019.
13. Neha Mahajan, and Vaishali Barkade, "Clustering Based Efficient Privacy Preserving Multi Keyword Search Over Encrypted Data", In 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), IEEE, pp. 1-6, 2018.
14. Xinrui Ge, Jia Yu, Chengyu Hu, Hanlin Zhang, and Rong Hao, "Enabling efficient verifiable fuzzy keyword search over encrypted data in cloud computing", IEEE Access, vol. 6, pp. 45725-45739, 2018.
15. Sanjit Chatterjee, Manish Kesarwani, Jayam Modi, Sayantan Mukherjee, Shravan Kumar Parshuram Puria, and Akash Shah, "Secure and efficient wildcard search over encrypted data", International Journal of Information Security, vol. 20, no. 2, pp. 199-244, 2021.
16. Kai He, Jun Guo, Jian Weng, Jiasi Weng, Joseph K. Liu, and Xun Yi, "Attribute-based hybrid Boolean keyword search over outsourced encrypted data", IEEE Transactions on Dependable and Secure Computing, vol. 17, no. 6, pp. 1207-1217, 2018.
17. Prasanthi Sreekumari, "Privacy-preserving keyword search schemes over encrypted cloud data: an extensive analysis", In 2018 IEEE 4th International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing,(HPSC) and IEEE International Conference on Intelligent Data and Security (IDS), IEEE, pp. 114-120, 2018.
18. Zhitao Guan, Xueyan Liu, Longfei Wu, Jun Wu, Ruzhi Xu, Jinhu Zhang, and Yuanzhang Li, "Cross-lingual multi-keyword rank search with semantic extension over encrypted data", Information Sciences, vol. 514, pp. 523-540, 2020.
19. Yuanbo Cui, Fei Gao, Yijie Shi, Wei Yin, Emmanouil Panaousis, and Kaitai Liang, "An efficient attribute-based multi-keyword search scheme in encrypted keyword generation", IEEE Access, vol. 8, pp. 99024-99036, 2020.
20. Cheng Guo, Ruhan Zhuang, Chin-Chen Chang, and Qiongqiong Yuan, "Dynamic multi-keyword ranked search based on bloom filter over encrypted cloud data", IEEE Access, vol. 7, pp. 35826-35837, 2019.
21. Yunyun Wu, Jingyu Hou, Jing Liu, Wanlei Zhou, and Shaowen Yao, "Novel multi-keyword search on encrypted data in the cloud", IEEE Access, vol. 7, pp. 31984-31996, 2019.
22. Hui Yin, Zheng Qin, Jixin Zhang, Lu Ou, Fangmin Li, and Keqin Li, "Secure conjunctive multi-keyword ranked search over encrypted cloud data for multiple data owners", Future Generation Computer Systems, vol. 100, pp. 689-700, 2019.
23. Hong Zhong, Zhanfei Li, Jie Cui, Yue Sun, and Lu Liu, "Efficient dynamic multi-keyword fuzzy search over encrypted cloud data", Journal of Network and Computer Applications, vol. 149, pp. 102469, 2020.
24. Jianfei Sun, Shengnan Hu, Xuyun Nie, and Joojo Walker, "Efficient ranked multi-keyword retrieval with privacy protection for multiple data owners in cloud computing", IEEE Systems Journal, vol. 14, no. 2, pp. 1728-1739, 2019.
25. Maryam Hozhabr, Parvaneh Asghari, and Hamid Haj Seyyed Javadi, "Dynamic secure multi-keyword ranked search over encrypted cloud data", Journal of Information Security and Applications, vol. 61, pp. 102902, 2021.

26. ZhangjieFu, Xingming Sun, Nigel Linge, and Lu Zhou, "Achieving effective cloud search services: multi-keyword ranked search over encrypted cloud data supporting synonym query", IEEE Transactions on Consumer Electronics, vol. 60, no. 1, pp. 164-172, 2014.

27. Tianyue Peng, Yaping Lin, Xin Yao, and Wei Zhang, "An efficient ranked multi-keyword search for multiple data owners over encrypted cloud data", IEEE Access, vol. 6, pp. 21924-21933, 2018.