Journal of Engineering Science and Technology Review 17 (4) (2024) 23-30

Research Article

JOURNAL OF
Engineering Science and Technology Review

www.jestr.org

Multi-Keyword Ranked Search in Cloud Environment

Vikas Sharma[1,*], Kapil Sharma[1], Akshi Kumar[2] and Pawan[1]

[1]*Department of Information Technology, Delhi Technological University, Delhi, India*
[2]*Department of Computing, Goldsmiths, University of London, United Kingdom*

Received 8 January 2024; Accepted 9 July 2024

Abstract

Nowadays, with the simplification of cloud storage complexity and expanding its capacity, the cloud has become more widely accessible. Data owners increasingly shift away from traditional systems to leverage cloud resources, enhancing capacity and geographic diversity. Despite these advancements, the need to safeguard confidential data prompts a transformation to encryption before outsourcing, establishing additional standards for data privacy. This shift renders conventional data utilization through standard keyword searching obsolete. Therefore, there is a crucial need for an encrypted cloud information search service. To address the challenge of privacy-preserving multi-keyword searches, this paper proposes an efficient solution that enhances the existing framework. The proposed system uses an advanced encryption standard (AES) to improve data security and management in a cloud environment. Each private server maintains a registered information file and deletes merged data from the free cloud after observation. Customers can manage order records in the cloud data warehouse by transparently logging observations. Access to server information requires client inspection requests and verification. Private servers validate capacity and issue tokens to the Key Distribution Center (KDC) for key distribution. Finally, the term frequency-inverse dense frequency (TF-IDF) equation is used for data positioning. This approach ensures secure, authenticated, and efficient data handling. This solution focuses on two key objectives: firstly, incorporating synonym-based search to facilitate multi-keyword synonym queries, and secondly, implementing a ranked search to yield more detailed search results. While existing search methods support only fuzzy keywords or precise queries over encrypted cloud data. The authors assess its performance by employing real-life datasets, and comprehensive experiments affirm the validity and practicality of the proposed scheme.

Keywords: Distributed Databases, Information Networks, Private Server, Public Cloud, Ranking Results

1. Introduction

To allow the centralized data repository and access to information services or resources whenever appropriate, the cloud provides a broad group of remote servers in a network [1]. Many IT companies and individuals are exporting their cloud server databases. Regardless of location, users can access and exchange information uploaded to the cloud. Outsourced data may contain very confidential information such as e-mails, financial information for companies, government documents, records of personal health care, and pictures of Facebook and business documents [2, 3].

Cloud service providers (CSPs) can access confidential information from users without authorization. CSPs' general strategy is to maintain the confidentiality of data under which data is encrypted until it is outsourced to cloud storage, which would affect the tremendous cost of data usability. To protect their privacy, data owners subcontracted their data to cloud servers in encrypted form in a secure quest for encrypted data. If a data user wants to search for a specific file, they send a keyword request to the cloud server. The cloud server then creates the most relevant data results for the data user. Safe keyword search over Sensitive information not only reduces the cost of computation and storage but also allows for a ranked search for multi-keywords, fuzzy keyword search, and search for similarities. Both of these systems are based on a single-ownership model. Previous work supports the single-owner model, in which the data owner must remain online to create data consumer trapdoors. Therefore, this paper suggests a multi-owner model to solve the limitations of earlier approaches, where multiple data owners store encrypted data and data owners remain online to create trapdoors simultaneously. To encrypt their secret data with different secret keys, various data owners exchange different secret keys. Safe search protocols are introduced in this paper, and cloud servers can conduct secure searches without knowing the actual value of keywords and trapdoors. In this multi-user and multi-owner cloud model, four private servers are involved. Apart from these private servers, one monitoring server collects the index from all the servers and merges them into an integrated main index. One key distribution server manages the key generation and distribution process. The cloud server stores the data received from the private server. Cloud server also provides the data to the data user upon proper authentication from the private server. Figure 1 shows the architecture of the system consisting of all these entities. Data owners create a stable, searchable index of the keyword set and extract keywords from files. The data owners submit a keyword index to the management server. Data owners encrypt files and outsource encrypted data to cloud servers. When the administration server sends an encrypted keyword index, the keyword index is re-encrypted by the administration server. The administration sends the server to an outside company for re-encryption, which outsources the cloud-stored index. To see if any files from the cloud server are open, the data user must first check if any appropriate traps have been set, which send data to the administration.

*E-mail address: vikassharma12387@gmail.com
ISSN: 1791-2377
doi:10.25103/jestr.174.04

After the user has completed their logon process, the system verifies their identity and stores it in the cloud as an encrypted secret key. If the cloud identifies the data owner (for example, the name and encryption key associated with the target file), it returns the top-K encrypted files. The data user downloads and decrypts the files from the cloud due to getting vast amounts of K data from the server. When a user receives top-N files from a cloud server, they are given N times the amount of data in decrypted files to download, and then they have to decrypt and unpack them. The data protection access file [4–6] and short-term data protection surveys and data exchange in the data organizations. The data protection agreements are applicants' agreements. PPI is a third-party catalog advantage (e.g., open cloud) that gives users or searchers global knowledge. A requester is involved in a two-part method for seeking stories of plots. Firstly, it includes a survey of the related phrases against the PPI server that summarises the applicant owners in the program (e.g., p0 and p1).

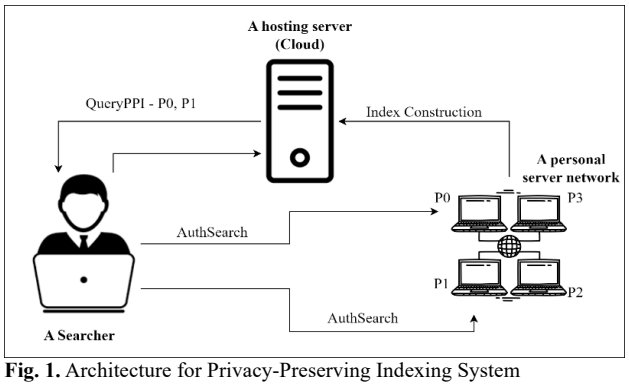

Fig. 1. Architecture for Privacy-Preserving Indexing System

The search administrator must then contact the server and customer reviews and permission requests for positive holders before local searches. At Rundown, a detailed breakdown. Remember that the data set is only validated and accepted but not on the Privacy-Preserving Indexing (PPI) server. It should be noted that PPI contrasts with current safe information efforts. [7-9] It is exemplary since 1) the data are straightforwardly placed on a PPI server (i.e., not encrypted), enabling the generation of valid, informed, and scalable data. PPI jams consumer privacy by climbing to clouds that could only be used through encryption. 2) Rough-grained data is supplied on the PPI server, while the first private content is left on the servers and handled by customer laws. The PPI system must protect various search queries and owners separately. The information indicates that each server has a different data structure and system with other words. Personal and confidential information should be protected under a PI if each party possessing it has more than one record of having substantial ownership. This strategy has significance in several ways. Firstly, it conceals the identities of all parties in private transactions. Secondly, it also hides the identity of individuals using the service. No words in such (single sentences) could be expanded. To take the point, for example, with an e-Health organization, the experience of a treatment associated with "preterm delivery" (e., undocumented instances of induction or even of difficult childbirth) is seen in an e. example) A language used to refer to a larger or smaller individual is more (or less) vulnerable than a word or phrase acting on its own. The respective appearances of content and motion are "solo" and "cont." For example, motion and content can be contained, and "cont." can be perceived as motion. When people are motivated by interest, they are less likely to form positions on divisive issues. Presenting the idea [6], [9] It provides expanded concepts about privacy concepts [expands ideas], which could be used to enable it to balance privacy and retention better. Data security measures of this good nature are too reasonable to use just once or for a short period, so it is almost impossible to use these programs that offer to say something quantitatively that they won't break. It's proposed that pi-MPPI, another PPI debate, regulates the spillage of data protection for a multi-watchword record look. This μ-MPPI framework paper describes the distinctive terms; value 0 is moderate in privacy security, while value 1 provides absolute privacy protection (including overhead for further inquiries). In other words, a multi-stage μ-MPPI aggressor can only guarantee that successful attacks have been conducted according to the label's privacy. It attempts to calculate the μ-MPPI in the data system's measuring points and frame contours. Č-development MPPIs require a cautious plan to legally shield false-positive people (i.e., any owner who's going to claim falsely to be without word and conspiracy) to maintain their privacy by protecting a genuinely positive owner from false-positive people.

A stable β-MPPI must be generated without exchanges in a proper knowledge organization of experts requiring mutual trust between self-regulatory operating servers. If a secure output spread is assigned, it is highly testable. To comply with the most stringent data protection standards in various multi-specific looks, μ-MPPI can be constructed as an improvement issue to fix complex calculations. In this paper, a multi-party system (MPC) is used to [10-12]. It maintains data protection and safeguards essential intelligence; existing MPCs can run virtually well in a restricted environment with a simple workload. For instance, FairplayMP is an effective MPC agent: it demands approximately 10 seconds to assess capacity. [13] Which is usually possible in milliseconds for reasonable non-safe calculations. Therefore, MPC procedures will lead to unbelievable and unsatisfactory costs if explicitly applied to the β-MPPI problem with skewed calculations and a wide range of servers. Our main goal is to align the secure and unsafe parts of the calculation to address the challenges of accurate and stable μ-PPI development. However, as predicted, we restrict the safe calculation portion by evaluating various techniques.

Here are the research gaps:

1) Ensuring robust encryption methods for data stored and managed in cloud environments.
2) Efficiently managing and retrieving large volumes of data in cloud systems while maintaining security.
3) Maintaining transparent and secure logging of data access and changes in cloud environments.
4) Implementing effective access control mechanisms to ensure only authorized users can access sensitive data.
5) Distributing and managing cryptographic keys securely in a cloud setting.
6) Develop efficient data retrieval and compression methods to optimize cloud storage and access speeds.

In this context, perplexing MPC NLP measurement has been effectively isolated to the extent. Its design convention μ-MPPI needs only a simple method of computing that enables the system to be implemented globally. The reference can be translated from this paper as follows.

- To address various needs, notably distinct privacy security concerns, we have created multi-term PPI. There is no question that financial planning is among the most

significant retirement planning topics. A model PI is instructed to look for non-existent pixels and control another to ensure no infringement co-occurs, guaranteeing privacy is protected quantitatively.

- For the network of widely untrusted servers, we have also proposed a set of MPPI software agreements. In particular, when exploring how many security responsibilities could be reshaped without abandoning the nature of privacy protection, single- and multi-term, the safety of the MPPI should be enhanced in line with the measurement model and the device configuration.
- We followed a working model of the MPPI, in which a test study reaffirms that our convention is a beneficial place for list construction.

2. Literature Review

The new knowledge networks [5] have an economic search for the documents distributed. PPIs maintain data security tables or indexes for your owner's privacy. This topic is understood given the existence of multiple-key emerging information networks [6] and privacy security. Data-protection indexes or PPIs protect the confidentiality of your owner. The known issue is privacy security when using PPI. Circumstances and phrases give them an inherited meaning. The author introduces the first e-PPI work for the quantitatively differentiated search for distributed records and the protection of their privacy. A stable, fuzzy, multi-keyword search for encrypted cloud information was suggested in [14]. This scheme allows the search parameter for multiple keywords and gives the related results—using coordination to calculate similarity based on safe internal product measurement. In [15], the author proposed a new method of attack and a strategy to resolve the identity-shattering issues associated with PPIs by implementing an extended PPI. With trusted third parties and trusted relationships between providers, the proposed e-PPI construction protocol has been initiated. The PPI construction protocol is introduced using a generic MPC technology, which secures multi-part computing and enhances performance to a realistic level by minimizing the costly MPC component. According to this research, the Low Weight Hash Tree is a low-maintenance query-efficient indexing technique [16]. A new naming system and description technique are used in tree form for the layout of the index. LIGHT has been developed with nearly optimal performance over a generic DHT system that supports numerous complex questions. Using ABE, Persona masks user information and sends users fine-grained policies to see their information. The data is hidden in the paper [17]. It provides practical applications where, by using a given privacy policy, users are not the OSN. This new cryptographic mechanism improves the general application of ABE. It describes an application that reproduces Facebook applications and provides appropriate results even on mobile devices when browsing privacy-enhanced web pages, contrasting both current and new and illustrates how Persona provides additional privacy benefits to the features of existing online social networks. This paper [18] demonstrates that hospitals and patients can use the device to exchange medical information with a third-party server. When accessing records from anywhere via a shared device is beneficial. The mutual protection of the system is essential. When medical records are encrypted with symmetrical encryption, authentication is applied, also known as private-key encryption; a sender sends encrypted data, and the recipient uses this form of encryption to decrypt the data.

A new data security index abstraction, SS-PPI, is proposed in [4] By the author, which provides theoretically guaranteed confidentiality protection in combination with distributed access managed search protocols. In contrast with current plans, our approach highlights various distinctive characteristics (for example, index flipping protection for data. [14]). (a) Includes access control measures that improve both search effectiveness and attack resistance in the Privacy Security Index, and (b) utilizes the Rapid Index Construct Protocol in a fully distributed way through the modern use of secret sharing. We implement two methods, formal and SS-PPI analysis, demonstrating that the latest privacy security and execution efficiency solutions are available.

Such confidence is gained in a place by making a proposal that eliminates the need for this authority. In a hierarchical multiuser control system, the approach maintains a centralized index of privacy preservation in line with the distributed access protocol. As the index has been made public, this new index has absolute privacy protections. To begin with, there is a twofold allure of a solution:

- Authors expect the service suppliers to monitor and ensure conformity with how groups are maintained in identifying and controlling who has access to the content.
- To compensate for their privacy and effectiveness issues, device developers minimize controls that can be manipulated.

No feature in this block is transmitted, but the machine compiles it into a circuit specification [6], which is not revealed in other blocks when calculating the Boolean result. Players cooperate with Fairplay, encouraging two competing entities to use resources efficiently and provide positive outcomes. Beaver's underlying FairplayMP protocol (and that's how they're usually described) is a constant number of touch counts (although, as you might have noted, they're generally referred to as FairplayMP) (8 rounds in our implementation). As an ongoing collaboration project to modernize the BMR to meet Ben-own Or's criteria, it has added new elements and substantially improved the existing methods. We ought to use this approach, as it is well acknowledged that the number of rounds applied in the process is crucial to the protocol's ultimate success [19]. To conduct job assignments while protecting them from the eyes of others, smaller organizations frequently need to exchange documents with one another. In this case, users need a document indexing facility that allows them to access documents easily (1) without exposing other document information, (2) as users, groups, and documentation change, and (3) without the need for users to settle on a single and fully trusted authority. In [5], the author proposes a concept of privacy that measures how much data is leaking from the index about something like the conditions contained in the inaccessible content. In addition, the system provides re-confidential indexing facilities with sensitive materials that use secret divisions and term mergers to set tenable limits for information leakage, even in the event of statistical attacks. The need for a confidential authority is removed in this document [6]. With a distributed search compliance access control protocol, the paper proposes a centralized PPI solution. This PPI, even when the index is written, guarantees strict confidentiality. This device has been tested in actual life experiments. The solution has two steps: firstly, service

providers maintain complete control when identifying access classes, and secondly, system implementers have control over the security and efficiency of their particular areas or document searches with PPI applications [20]. Circumstances and phrases give them an inherited meaning. The author introduces the first e-PPI work for the quantitatively differentiated search for distributed records and the protection of their privacy. This pursuit was suggested in [21] with symmetrical searchable encryption (SSE). They form a fluffy watchword collection of data with the help of distance of transition. He scans for Tw when the customer looks at CS and returns scratch-coordinating Tw papers. They compile the suitable fluffy phrase identified using a particular case and a fluffy Multiway Tree searchable list that uses an image-based trial to navigate the search.

3. Modules and Methodology

An open cloud server, several specialized servers, and several clients are included in the architecture. The owners' records are shared on private servers. Information is scratched throughout the construction.

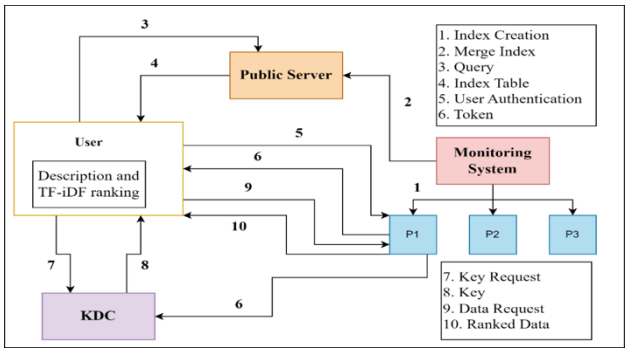

Fig. 2. System Architecture

Figure 2 illustrates the flow of data and interactions among various components involving a user, a public server, a monitoring system, and a Key Distribution Center (KDC). Here's a step-by-step breakdown of the process:

1) **Index Creation**: The Monitoring System indexes the data in various parts (P1, P2, P3).
2) **Merge Index**: These indexes are merged and sent to the Public Server.
3) **Query**: The user initiates a query to the Public Server.
4) **Index Table**: The Public Server responds with an index table relevant to the query.
5) **User Authentication**: The user authentication request is sent to the Public Server.
6) **Token**: A token is issued to the user, which likely serves as a session or access token for further requests.
7) **Key Request**: The user requests a key from the KDC.
8) **Key**: The KDC provides the key to the user.
9) **Data Request**: The user requests data from the Monitoring System.
10) **Ranked Data**: The Monitoring System provides ranked data to the user based on the query and the user's description and TF-IDF (Term Frequency-Inverse Document Frequency) ranking.

The diagram labels and arrows indicate the interactions and data flow between the components:

- **User**: Initiates the query, handles descriptions and TF-IDF rankings, and interacts with all other components.
- **Public Server**: Manages indexes and user authentication.
- **Monitoring System**: Manages data parts and index creation.
- **KDC (Key Distribution Center)**: Manages user authentication and data access keys.

AES Calculation is used for data encryption. Each private server has a registered information file. Observation is based on gathering and using all relevant material. Then, you will delete this merged file from the free cloud. A customer could place an order on the same day the order was sent or an "as-needed" order record in the cloud data warehouse. On the contrary, however, the transparent cloud records the observation phase as a single log. Query-related data is now available on this latest consolidation list for customers with a private server. To obtain access to all information on the server, the client must request an acceptable inspection with the headquarters and the watchword at this stage. This subtle object inventory is being checked by a private server (P1, P2, P3) in their database. After validating the current capacity, the personal server delivers the token to the Key Distribution Center (Key Distribution Center). When customers order keys, it should be done such that an update is applied. As KDC determines that this token is a private key from a public key, the search returns true. Once the KDC (Kerberosdale Database Server) has issued the customer an authentication key, the user's secret key will be expanded. The customer has sent the request to a private server that holds all scrambled documents in place and then compresses them into one stream. In theory, using the leading consumer needs clarification on the details; however, it is questionable whether it's a method that can yield an adequate result. In the end, use the TF-IDF positioning equation to obtain the effects of your positioning setup.

A system consisting of the following modules:

- **Deployment of System**
  AES Client Side Interface Encryption and Decryption Register and log in to the link programming and data exchange database, client, and server.
- **Creation algorithm for MPPI Index**
  When you use the MPPI formula to collect a list of all personal servers, you create a global infrastructure. Since there is no way to make this list more complete and there is no risk of keeping user data confidential, the full list requires reviewing all their data.
- **Uploading the combined index over Public Server**
  When the control system has finished assigning each server a private cloud with the consolidated version of the latest update file, it is the responsibility of that private cloud to pass the whole file to the rest of the others.
- **Receiving response based on input query From Public Server**
  This is a cloud service request that can handle only one piece of data, a cloud file created on the private server for use only by this user.
- **Token Generation and Authentication of Client**

To access the private server after the data has been obtained, the client must establish a connection to retrieve the data. If the user's identity has been authenticated, the client connects to the server and distributes tokens to designated Key Distribution Center (KDC) nodes, and the Key Distribution Service starts.

- **Distribution of Keys and Decrypting File**
  KDC does the work for the client, whoops, by handing the private servers' values into tokens so they can be decoded.
- **Ranking Results based on TF-IDF**
  Following confirmation, the customer receives private server scrambling reports. These shattered performances are then unbundled using the obtained KDC key. Finally, by using TF IDF, you create the positioning.

4. Mathematical Formulation

Consider the System to be represented by S.

S= {Ω, P, O}

where,

- Input Ω: The system takes a multi-keyword Query as an input
- Output (O): Ranking.
- Process (P)

a) Publication of single-term index

$$\epsilon_j = \frac{(1-\sigma_j).\beta_j(t_j)}{(1-\sigma_j).\beta_j+\sigma_j} \quad (1)$$

$$\beta_j = \left[(\sigma_j^{-1}-1)(\epsilon_j^{-1}-1)\right]^{-1} \quad (2)$$

Where $\epsilon_j$stands for single-term index, and βj stands for all the possible values that are derived from source analysis [22].

b) Rate of False Positive:
FP (0; 1) = F (0; 1)

$$FP(0,1) = \frac{F(0,1)}{F(0,1)+\sigma_0\sigma_1} \quad (3)$$

The most pertinent probability of an owner who is non-positive but publishes the details as a positive owner is β0; β1 When FP(0,1) is the false positive value [22].

c) Index Production
Ω = {Ω1, Ω2... Ωn}
Where Ω represents the set of indexes collected from all the servers.

d) Merging the index of all the Private Servers and uploading.
M = {M1, M2... Mn}
Where M is the collection of all the merge indices the monitoring system has obtained.

e) User Query to public server
θ= {θ1, θ2... θn}
here, θ represents the set of cloud queries from a public server.

f) Authentication of User at private server
μ= {μ1, μ2... μn}
here, μ refers to authenticated users on a private server.

g) Distribution and Generation of Tokens
δ= {δ1, δ2... δn}
here, δ is the collection of tokens for authenticated users created by a private server.

h) KDC for Key Generation
K= {K1, K2... Kn}
where K is the set of KDC keys used for user-side data decryption.

i) Data decryption and TF IDF ranking
Φ= {Φ1, Φ2... Φn}
here, Φ is the collection of all outcomes for the particular input query.

5. Algorithms

a) Advanced Encryption Standard (AES) Algorithm:
The AES is a two-encrypted substitution cipher of 128 bits—separate 3 Expansion three AES keys with 128 or 192-bit and 256-bit lengths. In addition to using square keys, the encryption method employs a specific configuration called round keys. AES is more of an iterative process than a stream cipher like Feistel. Increasing the number of keys by 12 bits requires 16 additional pieces, and expanding the number of fragments by 24 bits requires 18 additional vital positions. In general, only 128-bit components are used in computer memory expansions. On each turn, each player can trade one or more bytes, add one line commentary and one new character/character of one or more lines, and make an addition and removal. These four measures are handled differently depending on whether you want to expand or reduce the size. The encryption and decryption contain the following steps:

Steps for Encryption:

- Substitution of Bytes
- Shift rows
- Mix Columns
- Add round key

Steps for Decryption:

- Add round key
- Mix columns
- Shift rows
- Byte substitution

b) Term Frequency (TF) — Inverse Dense Frequency (IDF):
This method calculates how central a word is to a text collection by determining how often it appears in a particular type frequency section. Expansion: TF-IDF refers to the number of words and occurrences of each one of those words in the document, but getting information about the papers that exhibit TF is very closely correlated with these is nearly the same as discovering words in context. Because every document is different in length, there are far more words in documents that appear far longer than in short ones.

$$tf(t,d) = \frac{f_d(t)}{\max f_d(w)} \quad (4)$$

After calculating the TF values for all terms chosen, the index is extended with the five highest-scoring terms, which produces the top 5 terms in the index. For index creation and table construction, a table is needed to build and add the keyword(s). As will be "filename" and the size of the index "keyword" as a metadata field, the table will contain it when it is formed. This table is sent to the reporting server for further processing and then sent to the final results server.

$$idf(t,D) = In\left(\frac{|D|}{|\{d \in D : t \in d\}|}\right) \quad (5)$$

IDF: Frequency of the inverse text, measuring the importance of a term. During TF computing, all words are considered equally essential. However, it is understood that such terms, such as 'is' and 'of' and 'that,' may often occur but are of little value. Therefore, by measuring the following, we can measure down the periodic terms while analyzing the uncommon ones:

c) Iterative-Publish (Owner Pi, set β_0 (r_k))
for all k ϵ [0; l -1] do β' (r_k) is topologically sorted
if match (cur-memvec, getStartingState(r_k))
then
β_{cur}<mem_{vec}
where the current membership vector cur-member publish
(cur-member, β' (r_k))
end if
end for

We advocate using the Iβ method to merge phrases to provide a broader range of probabilities in our production. The Index Method Shows how iteratively, sentence by sentence, page by page, the indexing method functions.

6. Result and Discussion

The proposed method applies a two-way encryption model for data queries, so the data remains complete and undistributed. Two-way encryption would protect user privacy. When we are measuring time, we have to consider some additional things, such as (a) file upload time, (b) search time, (c) encryption time, and (d) the time to generate tokens.

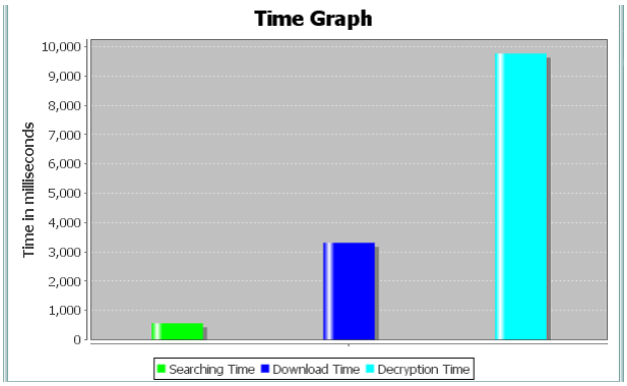

Fig. 3. Time Graph at User Side

Similarity Measurement

In this table, there are two different systems: the one that already exists and the second that you are thinking of creating with the help of similarity indexes. This model implemented four iterations successfully, and every iteration gave a different, more significant result than the previous one.

Table 1. Similarity Table

	Existing	Proposed
D1	0.47	0.93
D2	0.78	0.95
D3	0.38	0.98
D4	0.47	0.97

Now, you can quickly analyze the proposed method, which is more similar to the reference system (Graph) for the four texts in Figure 4.

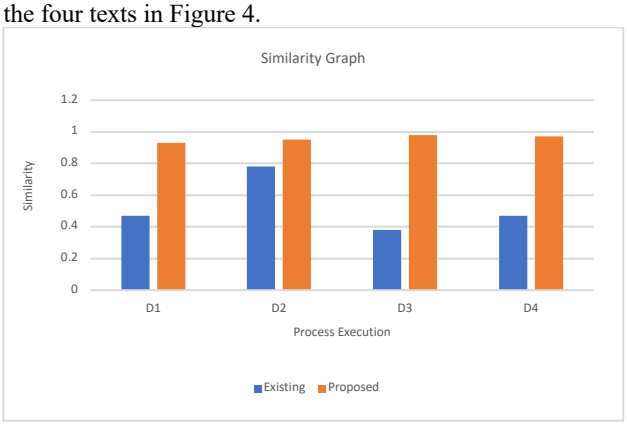

Fig. 4. Time Graph for Process Execution

Time Measurement

You can see that each operation generally takes the same time if you extend the chart to show time vs. different activities like uploading, indexing, querying, and token generation. Work on the project twice, gather data, and plot your findings.

Table 2. Time Measurement Table

	File Upload	Search Query	Time for Encryption	Time for Token Generation	Ranking
D1	3.08	0.92	3.40	0.37	0.88
D2	5.84	0.47	2.60	0.48	0.78

The data analysis in Figure 5 demonstrates that the suggested system would follow the trajectory over time. Find the table cell B2 from the chart and add its value to the graph.

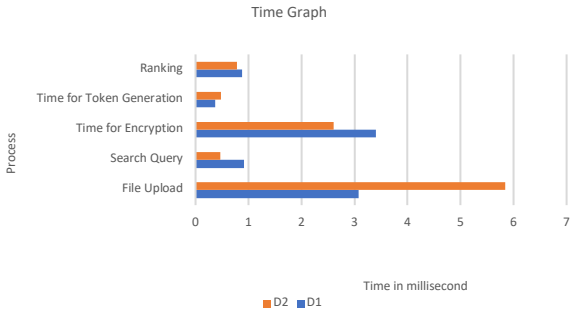

Fig. 5. Time Graph

Figure 5 illustrates that the proposed system uses less memory since it distributes the Key Distribution Center (KDC). The x-axis shows the number of main requests made, and the y-bar shows the memory used.

The implementation of a two-way encryption model for data queries, as discussed, ensures data integrity and enhances user privacy. This model's scalability and adaptability are critical for its practical application in cloud environments.

Scalability Issues

1. **Resource Utilization:** The proposed system's efficiency in terms of memory usage, as shown in Figure 5, indicates that it can handle an increasing number of requests without a proportional increase in resource consumption. This is due to the distributed Key Distribution Center (KDC), which decentralizes the processing load, thus preventing bottlenecks typically encountered in centralized systems.
2. **Performance Overhead:** Despite the added steps of encryption and token generation, the performance overhead remains manageable. Table 2 illustrates that the time required for encryption and token generation does not significantly degrade the system's responsiveness, even as the file upload and search query times vary. For large-scale applications, maintaining this balance is crucial for ensuring user satisfaction and operational efficiency.

Adaptability Issues

1. **Integration with Existing Systems:** Integrating the proposed model with existing cloud systems involves minimal disruption due to its compatibility with standard data querying processes. The similarity measurement improvements (Table 1) suggest that the model can be effectively applied to various datasets, enhancing the accuracy and reliability of data retrieval.
2. **Dynamic Workloads:** Cloud applications often experience fluctuating workloads. The model's ability to handle different file sizes and query complexities, as shown in the time measurement analysis (Figure 4), demonstrates its adaptability. The consistent performance across various scenarios (D1, D2, etc.) underscores its robustness in dynamic environments.

Comparative Analysis with Current Technologies

1. **Security Enhancements:** Current cloud encryption technologies often focus on either data-at-rest or data-in-transit encryption. The two-way encryption model uniquely combines these aspects, providing continuous protection from the point of data upload to query processing. This holistic approach significantly reduces vulnerabilities compared to traditional single-layer encryption methods.
2. **Efficiency in Data Retrieval:** The similarity indexes used in the proposed model (Table 1) outperform existing systems in terms of accuracy and relevance of query results. By refining data retrieval processes, the model speeds up query responses and ensures higher precision, which is vital for real-time data analytics applications.
3. **User Privacy:** Enhancing user privacy through continuous encryption significantly improves current technologies that may expose data during processing. The proposed model's end-to-end encryption mitigates such risks, aligning with stringent data protection regulations (e.g., GDPR, CCPA).

Broader Implications

1. **Regulatory Compliance:** The enhanced privacy features make the system highly attractive for industries with strict regulatory requirements, such as healthcare and finance. Organizations can better comply with data protection laws by ensuring data remains encrypted throughout its lifecycle.
2. **Future Innovations:** The success of the proposed model paves the way for future innovations in cloud security. For instance, combining two-way encryption with machine learning algorithms could enhance data protection while optimizing performance.
3. **Economic Impact:** By reducing the resource requirements for secure data processing, the model can lower operational costs for cloud service providers. This cost-efficiency can be passed on to consumers, making secure cloud services more accessible and affordable.

By addressing both practical and broader implications, the model demonstrates its potential to revolutionize cloud data management and set new standards in the industry.

7. Conclusions

In conclusion, this paper has delved into the transformative impact of cloud computing on medical education, emphasizing its pivotal role in reshaping learning experiences for aspiring healthcare professionals. The integration of cloud technologies promises a future characterized by enhanced connectivity, accessibility, and preparedness to tackle the evolving challenges of modern healthcare. Moreover, as the complexity of cloud storage diminishes and its capacity expands, a paradigm shift is observed, with cloud resources becoming increasingly appealing. However, the critical aspect of ensuring data privacy is addressed through the proposed encryption system, which requires access to sensitive information or data for thorough validation. The paper also recommends the utilization of more settled figures and implementing a secure way for participating students to increase the percentage of available caution in the MPC nursing course. To facilitate this, the platform aims to establish a link between the neighbourhood and cloud servers, allowing seamless information sharing among customers. The encryption system plays a crucial role in handling validation and ensuring the security of sensitive data. As a concrete step, the paper encourages interested parties to initiate the process in a dedicated section, providing their details through a form and selecting the proposed system for implementation. This comprehensive exploration and offered solutions underscore the potential of technology, particularly cloud computing, in shaping the future of medical education, fostering innovation, and preparing the next generation of healthcare professionals for success in an ever-evolving healthcare landscape.

Future work in this domain could focus on enhancing the scalability and flexibility of encrypted cloud information search services to accommodate diverse user needs and dynamic changes in data storage requirements. Additionally, there is scope for further research into optimizing the efficiency and security of these services by integrating emerging technologies and collaborating with regulatory bodies to ensure compliance with evolving data privacy regulations.

References

[1] V. Sharma, K. Sharma, and A. Kumar, "From Theory to Practice: A Systematic Review of Digital Twin Implementations Across Industry 4.0," in *2023 14th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT),* Delhi, India: IEEE, Jul 2023, pp. 1–7, doi: 10.1109/ICCCNT56998.2023.10308052.

[2] V. Sharma and S. Ramamoorthy, "A review on secure data access through multi-keyword searching in cloud storage," in *Proc. 3rd Int. Conf. Intell. Commun. Technol. Virtual Mob. Networks, (ICICV),* Tirunelveli, India: IEEE, Mar. 2021, pp. 70–73, doi: 10.1109/ICICV50876.2021.9388595.

[3] B. Cheng, L. Zhuo, Y. Bai, Y. Peng, and J. Zhang, "Secure index construction for privacy-preserving large-scale image retrieval," in *Proc. - 4th IEEE Int. Conf. Big Data Cloud Comput. (BDCloud,* Sydney, NSW, Australia: IEEE, Dec. 2014 , pp. 116–120, doi: 10.1109/BDCloud.2014.36.

[4] M. A. Kamoona and A. M. Altamimi, "Cloud E-health Systems: A Survay on Security Challenges and Solutions," in *2018 8th Int. Conf. Comput. Sci. Inf. Technol. (CSIT)* Amman, Jordan: IEEE, Oct. 2018, pp. 189–194, doi: 10.1109/CSIT.2018.8486167.

[5] S. Chenthara, K. Ahmed, H. Wang, and F. Whittaker, "Security and Privacy-Preserving Challenges of e-Health Solutions in Cloud Computing," *IEEE Access*, May 2019 vol. 7, pp. 74361–74382, doi: 10.1109/ACCESS.2019.2919982.

[6] M. Dawoud and D. T. Altilar, "Cloud-based e-health systems: Security and privacy challenges and solutions," in *2nd Int. Conf. Comput. Sci. Eng. (UBMK)* Antalya, Turkey: IEEE, Nov. 2017, pp. 861–865, doi: 10.1109/UBMK.2017.8093549.

[7] M. Du, Q. Wang, M. He, and J. Weng, "Privacy-Preserving Indexing and Query Processing for Secure Dynamic Cloud Storage," *IEEE Trans. Inf. Forensics Secur.*, Mar. 2018, vol. 13, no. 9, pp. 2320–2332, doi: 10.1109/TIFS.2018.2818651.

[8] S. Almakdi and B. Panda, "Secure and Efficient Query Processing Technique for Encrypted Databases in Cloud," in *Proc. - 2019 2nd Int. Conf. Data Intell. Secur. (ICDIS) S*outh Padre Island, TX, USA: IEEE, Oct. 2019, pp. 120–127, doi: 10.1109/ICDIS.2019.00026.

[9] R. Zhang, C. Chow, and L. C. K. Hui, "Crypt-EHRServer: Protecting confidentiality with attribute-based encryption and encrypted query processing," in *Proc. - 14th Int. Symp. Pervasive Syst. Algorithms Networks, I-SPAN 2017, 11th Int. Conf. Front. Comput. Sci. Technol. (FCST) 2017 3rd Int. Symp. Creat. Comput. (ISPAN-FCST-ISCC)* Exeter, UK: IEEE, Nov. 2017, pp. 234–241, doi: 10.1109/ISPAN-FCST-ISCC.2017.56.

[10] H. Dai, Y. Ji, G. Yang, H. Huang, and X. Yi, "A Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Data in Hybrid Clouds," *IEEE Access*, Dec. 2019, vol. 8, pp. 4895–4907, doi: 10.1109/ACCESS.2019.2963096.

[11] D. D. Rane and V. R. Ghorpade, "Multi-user multi-keyword privacy preserving ranked based search over encrypted cloud data," in *2015 Int. Conf. Pervasive Comput. Adv. Commun. Technol. Appl. Soc.* (ICPC) Pune, India: IEEE, Apr. 2015, pp. 1–4, doi: 10.1109/PERVASIVE.2015.7087044.

[12] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," *IEEE Trans. Parallel Distrib. Syst.*, Nov. 2013, vol. 25, pp. 222–233, doi: 10.1109/TPDS.2013.45.

[13] N. Chandran, D. Gupta, A. Rastogi, R. Sharma, and S. Tripathi, "EzPC: Programmable and efficient secure two-party computation for machine learning," in *Proc. - 4th IEEE Eur. Symp. Secur. Privacy, (EUROSP)* Stockholm, Sweden: IEEE, Aug. 2019, pp. 496–511, doi: 10.1109/EuroSP.2019.00043.

[14] M. Hastings, B. Hemenway, D. Noble, and S. Zdancewic, "SoK: General purpose compilers for secure multi-party computation," in *Proc. - IEEE Symp. Secur. Priv.*, (SP) San Francisco, CA, USA*:* IEEE*,* May 2019, pp. 1220–1237, doi: 10.1109/SP.2019.00028.

[15] J. Bringer, H. Chabanne, and A. Patey, "Privacy-preserving biometric identification using secure multiparty computation: An overview and recent trends," *IEEE Signal Process. Mag.*, Feb. 2013, vol. 30, no. 2, pp. 42–52, doi: 10.1109/MSP.2012.2230218.

[16] P. Tang, R. Chen, S. Su, S. Guo, L. Ju, and G. Liu, "Differentially private publication of multi-party sequential data," in *Proc. - Int. Conf. Data Eng.*, (ICDE) Chania, Greece*:* IEEE*,* Apr. 2021, pp. 145–156, doi: 10.1109/ICDE51399.2021.00020.

[17] P. Derbeko, S. Dolev, E. Gudes, and J. D. Ullman, "Efficient and Privacy Preserving Approximation of Distributed Statistical Queries," *IEEE Trans. Big Data*, Jan. 2021, vol. 8, no. 5, pp. 1399–1413, doi: 10.1109/TBDATA.2021.3052516.

[18] I. Gupta, A. K. Singh, C. N. Lee, and R. Buyya, "Secure Data Storage and Sharing Techniques for Data Protection in Cloud Environments: A Systematic Review, Analysis, and Future Directions," *IEEE Access*, Jul. 2022, vol. 10, pp. 71247–71277, doi: 10.1109/ACCESS.2022.3188110.

[19] G. Zhao, C. Rong, J. Li, F. Zhang, and Y. Tang, "Trusted data sharing over untrusted cloud storage providers," in *Proc. - 2nd IEEE Int. Conf. Cloud Comput. Technol. Sci., (CloudCom)* Indianapolis, IN, USA: IEEE, Dec. 2010, pp. 97–103, doi: 10.1109/CloudCom.2010.36.

[20] J. Wei, W. Liu, and X. Hu, "Secure data sharing in cloud computing using revocable-storage identity-based encryption," *IEEE Trans. Cloud Comput.*, Mar. 2016, vol. 6, no. 4, pp. 1136–1148, doi: 10.1109/TCC.2016.2545668.

[21] B. S. Rawal and S. S. Vivek, "Secure Cloud Storage and File Sharing," in *Proc. - 2nd IEEE Int. Conf. Smart Cloud, (SmartCloud)* New York, NY, USA: IEEE, Nov. 2017, pp. 78–83, doi: 10.1109/SmartCloud.2017.19.

[22] A. Kumar, B. G. Lee, H. Lee, and A. Kumari, "Secure storage and access of data in cloud computing," in *Int. Conf. ICT Converg.*, (ICTC) Jeju, Korea*:* IEEE*,* Oct. 2012, pp. 336–339, doi: 10.1109/ICTC.2012.6386854.

[23] S. Sun, H. Ma, Z. Song, and R. Zhang, "WebCloud: Web-Based Cloud Storage for Secure Data Sharing Across Platforms," *IEEE Trans. Dependable Secur. Comput.*, Nov. 2020, vol. 19, no. 3, pp. 1871–1884, doi: 10.1109/TDSC.2020.3040784.

[24] M. Ali *et al.*, "SeDaSC: Secure Data Sharing in Clouds," *IEEE Syst. J.*, Jan. 2015, vol. 11, no. 2, pp. 395–404, doi: 10.1109/JSYST.2014.2379646.

[25] M. Akil, L. Islami, S. Fischer-Hübner, L. A. Martucci, and A. Zuccato, "Privacy-preserving identifiers for IoT: A systematic literature review," *IEEE Access*, Sep. 2020, vol. 8, pp. 168470–168485, doi: 10.1109/ACCESS.2020.3023659.

[26] P. Tiago, N. Kotilainen, and M. Vapa, "Mobile search - Social network search using mobile devices demonstration," in *2008 5th IEEE Consum. Commun. Netw. Conf. (CCNC)* Las Vegas, NV, USA: IEEE, Jan. 2008, pp. 12-45, doi: 10.1109/ccnc08.2007.290.

[27] L. Cherkasova and S. R. Ponnekanti, "Optimizing a 'content-aware' load balancing strategy for shared web hosting service," in *Proc. - IEEE Comput. Soc. Annu. Int. Symp. Model. Anal. Simul. Comput. Telecommun. Syst. (MASCOTS)*, San Francisco, CA, USA*:* IEEE*,* Sep. 2000, pp. 492–499, doi: 10.1109/MASCOT.2000.876576.

[28] S. Han, K. Han, and S. Zhang, "A Data Sharing Protocol to Minimize Security and Privacy Risks of Cloud Storage in Big Data Era," *IEEE Access*, May 2019, vol. 7, pp. 60290–60298, doi: 10.1109/ACCESS.2019.2914862.

[29] R. Li, C. Shen, H. He, X. Gu, Z. Xu, and C. Z. Xu, "A Lightweight Secure Data Sharing Scheme for Mobile Cloud Computing," *IEEE Trans. Cloud Comput.*, Jan. 2017, vol. 6, no. 2, pp. 344–357, doi: 10.1109/TCC.2017.2649685.

[30] Y. Tao, P. Xu, and H. Jin, "Secure Data Sharing and Search for Cloud-Edge-Collaborative Storage," *IEEE Access*, Dec. 2019, vol. 8, pp. 15963–15972, doi: 10.1109/ACCESS.2019.2962600.

[31] Y. Yang and Y. Zhang, "A generic scheme for secure data sharing in cloud," in *Proc. Int. Conf. Parallel Process. Work.*, (CPPW) Taipei, Taiwan*:* IEEE*,* Sep. 2011, pp. 145–153, doi: 10.1109/ICPPW.2011.51.

[32] V. Swathy, K. Sudha, R. Aruna, C. Sangeetha, and R. Janani, "Providing advanced security mechanism for scalable data sharing in cloud storage," in *Proc. Int. Conf. Inven. Comput. Technol. (ICICT)* Coimbatore, India: IEEE, Jan. 2017*,* pp. 1–6, doi: 10.1109/INVENTIVE.2016.7830237.