



International Journal of Advanced Research in Education and TechnologY (IJARETY)

Volume 12, Issue 3, May-June 2025

Impact Factor: 8.152



Privacy-Preserving Deduplication for Textual Data in Cloud Environments

Shaik Pujitha¹, Veeranki Jaya Shankar Bhavani², Vavalas Shivaram³, Vara Shivaram⁴

Assistant Professor, Department of CSE, Guru Nanak Institute of Technology, Hyderabad, Telangana, India¹

Student, Department of CSE, Guru Nanak Institute of Technology, Hyderabad, Telangana, India^{2, 3, 4}

ABSTRACT: The exponential growth of textual data in Vision-and-Language Navigation tasks presents significant challenges for data management in large-scale storage systems. Data deduplication has emerged as a practical strategy for data reduction in large-scale storage systems, but it has also raised security concerns. This project introduces DEDUCT, an innovative data deduplication method for textual data. DEDUCT employs a hybrid approach that combines cloud-side and client-side deduplication mechanisms to achieve high compression rates while maintaining data security. DEDUCT's lightweight pre-processing and client-side deduplication make it suitable for resource-constrained devices like IoT devices. This substantial reduction in storage demands can lead to significant cost savings and improved efficiency in large-scale data management systems. This can significantly reduce bandwidth consumption by sending only unique data. However, client-side deduplication raises concerns regarding data leakage. This hybrid approach optimizes data storage, enhances performance, and safeguards data confidentiality in various applications and other domains.

KEYWORDS: Cloud Service Provider, Public Cloud, Private Cloud, Encryption, Tokenization, Key Distribution Center.

I. INTRODUCTION

Vision-and-Language Navigation (VLN) [1] tasks are becoming increasingly important due to their significant impact on advancing autonomous vehicles and intelligent systems. VLN technology empowers agents to navigate real world environments, enhancing human-robot interactions and safeguarding safety in autonomous vehicle operations. Beyond navigation, VLN applications extend to diverse domains, including robotics, virtual assistants, and smart homes, making human-machine interactions more intuitive and user-friendly. The significance of textual data in VLN cannot be overstated, as it is the foundation for communication between humans and autonomous agents. Users convey detailed navigational commands through natural language instructions, and autonomous systems rely heavily on the accurate interpretation and execution of these textual directives. Efficient data management has become critical to meet the increasing demands of VLN and its associated applications.

Data deduplication [2] is a highly effective technique for reducing storage space consumption by eliminating the need for storing identical files or data blocks multiple times. Instead, only one copy of each unique data is stored, and references are used to point to the original copy. This method is particularly beneficial in cloud environments where vast amounts of data are typically stored. In backup applications, deduplication can reduce storage needs by up to 90–95% [5], while in standard file systems, it can lead to a reduction of up to 68% [4].

There are three main categories of data deduplication techniques based on granularity [53]: file level, fixed-size block, and variable-sized block. File-level deduplication finds and removes entire duplicate files. Fixed-size block deduplication divides a file into fixed-size blocks and eliminates duplicate blocks. Variable-sized block deduplication utilizes various sizes of chunks to identify redundant data, but it may create more metadata and lead to hash collisions. Block level deduplication is typically more efficient as it can detect duplicates even if they are stored across different files or portions of the storage system. Deduplication techniques can also be categorized based on place: server-based and client-based. Server-based deduplication identifies and eliminates duplicate data on the server. Server-based deduplication eliminates the need for users to perform deduplication tasks locally. However, server-side deduplication may only partially mitigate communication overhead. On the other hand, client-side deduplication takes place on the user's device before uploading data to the cloud. It involves collaboration between the client and server to find redundant data. This can significantly reduce bandwidth consumption by sending only unique data. However, client-side deduplication raises concerns regarding side-channel attacks and data leakage. Finally, deduplication can be

classified based on time: inline and offline. Inline deduplication eliminates duplicate data before or as it is being stored. Offline deduplication deals with deduplication after data is stored on a storage device.

Classic Deduplication (CD) methods [9] primarily focus on identifying and removing duplicate files, which can lead to inefficient storage when files share similar content but are not identical. Generalized Deduplication (GD) has emerged as a more comprehensive approach to address this limitation. GD expands the scope of traditional methods by recognizing and eliminating nearly identical or similar data chunks. This reduces storage requirements significantly, eliminates data redundancy, and improves data management efficiency.

II. LITERATURE SURVEY

Abadi et al. (2024) address the challenge of redundant data in federated learning (FL), especially in training large language models (LLMs), in their paper "Privacy-Preserving Data Deduplication for Enhancing Federated Learning of Language Models". In FL, multiple clients contribute local datasets, leading to overlapping or duplicate text samples that can hinder training efficiency and model performance. To mitigate this, the authors propose the Efficient Privacy-Preserving Multi-Party Deduplication (EP-MPD) protocol. EP-MPD performs deduplication before model training, reducing unnecessary computation and communication overhead. Notably, it operates without requiring a trusted central server, making it suitable for adversarial or semi-honest environments. The protocol utilizes two novel variants of the Private Set Intersection (PSI) protocol to ensure privacy. Experimental results demonstrate that EP-MPD not only maintains data privacy but also significantly enhances training outcomes, with up to a 19.61% improvement in perplexity and up to a 27.95% reduction in running time.

Z.Wang, W. Gao, M. Yang, and R. Hao (2023) explore critical challenges in medical data sharing within cloud-assisted electronic medical systems. Their study, "Enabling Secure and Efficient Data Sharing in Cloud-Assisted Electronic Medical Systems is Achieved Through Data Deduplication and Sensitive Information Protection," focuses on addressing redundant data storage and patient privacy risks. They propose a secure data sharing scheme that integrates data deduplication with sensitive information protection, replacing sensitive details with wildcards before encryption. This method enhances both privacy and deduplication efficiency, allowing authorized researchers to access decrypted records while keeping sensitive data concealed. By categorizing diagnostic information based on duplication rates, the scheme enables selective data downloads. Furthermore, it is designed to withstand brute-force and single-point-of-failure attacks, with experimental results demonstrating superior efficiency compared to existing solutions.

K.Ghassabi, P. Pahlevani, and D. E. Lucani (2023) explore the role of natural language processing (NLP) in enhancing data deduplication for large-scale systems, particularly focusing on textual data. Their paper, "Deduplication of Textual Data by NLP Approaches," addresses the limitations of traditional methods that only detect identical data, proposing a more advanced solution called TL-GD. This generalized deduplication method improves cloud storage efficiency by identifying similar—not just identical—data chunks. TL-GD operates by segmenting textual data into smaller components, which are then transformed into bases and deviations, allowing for more effective storage optimization. The method was tested on two real-world datasets and demonstrated nearly 67% lossless compression on textual navigation instruction data, yielding an average 25% improvement over conventional deduplication techniques.

P.Prajapati and P. Shah (2022) present a comprehensive review of techniques for ensuring secure data deduplication in cloud storage environments in their paper, "A Review on Secure Data Deduplication: Cloud Storage Security Issue." Cloud service providers optimize storage and bandwidth by eliminating redundant copies through deduplication, but this raises concerns regarding security, privacy, integrity, and confidentiality for clients. While traditional encryption methods enhance security, they often hinder deduplication efficiency. To address this challenge, the authors discuss Convergent Encryption (CE) combined with Proof of Ownership (PoW), enabling both security and deduplication. They further explore alternative techniques such as Provable Data Possession (PDP), Proof of Retrievability (POR), secure keyword search, DupLESS, Proof of Storage with Deduplication (PoSD), Dekey, Message-Locked Encryption, Attribute-Based Encryption (ABE), and Identity-Based Encryption (IBE). Their study assesses these methods, highlighting their effectiveness in maintaining security while ensuring optimized storage in cloud systems.

H.Sehat, A. L. Kloborg, C. Mørup, E. Pagnin, and D. E. Lucani (2022) examine the concept of dual deduplication to enhance data compression in cloud storage while preserving client privacy in their paper, "Bonsai: A Generalized Look at Dual Deduplication." As Cloud Service Providers (CSPs) offer vast storage at low costs, user data privacy remains a key concern. Dual deduplication allows clients to perform lightweight, information-theoretic transformations

before uploading data, ensuring CSPs cannot access raw content while still enabling deduplication. The authors introduce Bonsai, an advanced dual deduplication method designed to reduce storage fingerprints and enhance scalability. Bonsai provides notable benefits, including reduced client-side storage, minimized total storage for both clients and CSPs, and faster deduplication processing on the CSP's end. Experimental results show Bonsai achieving 68% compression on the cloud and 5% on the client side, allowing effective duplicate detection. Furthermore, when combined with universal compressors like Brotli, Bonsai enhances overall compression beyond using either approach alone. The study also highlights Bonsai's strong privacy guarantees against honest-but-curious CSPs that may have prior knowledge of clients' data distributions.

III. EXISTING SYSTEM

Traditional data deduplication methods solely focus on identifying exact duplicates. However, this approach may not be optimal for scenarios where data chunks share significant similarities but are not identical. Generalized Deduplication (GD) has emerged as a more comprehensive technique to address this limitation. This approach includes a transformation step, transforming each data chunk into a base and a deviation. The objective is to recognize similarities within the data, such as chunks sharing the same base, to achieve greater compression potential. The base value is pivotal in deduplication, while the deviation value captures the differences between the original data chunk and the extracted base.

Existing System Disadvantages

- Less performance.
- Public approach data storage.
- Less data confidentiality client and server storage side.

Proposed System

- We propose DEDUCT (Deduplication for Cloud Text), a deduplication method explicitly designed for textual data. DEDUCT builds upon the framework presented, emphasizing data security and optimization of storage efficiency.
- Client-side deduplication takes place on the user's device before uploading data to the cloud. It involves collaboration between the client and server to find redundant data. This can significantly reduce bandwidth consumption by sending only unique data. However, client-side deduplication raises concerns regarding side-channel attacks and data leakage.
- This hybrid approach optimizes data storage, enhances performance, and safeguards data confidentiality in various applications and other domains.

Proposed System Advantages

- Providing enhances performance.
- Hybrid approach optimizes data storage
- Improve more security and Data confidentiality client and server storage side.

IV. SYSTEM ARCHITECTURE

This cloud-based system is designed to ensure data security at every stage, from authentication and file upload to processing and result display. It actively detects potential threats while managing access through token-based authentication and private key generation, making data retrieval secure. With both public and private cloud segments working together, the system balances accessibility with strong encryption. Whether users access their data with or without key encryption, they can trust that cybersecurity measures are in place to safeguard against unauthorized access while maintaining smooth and efficient processing.

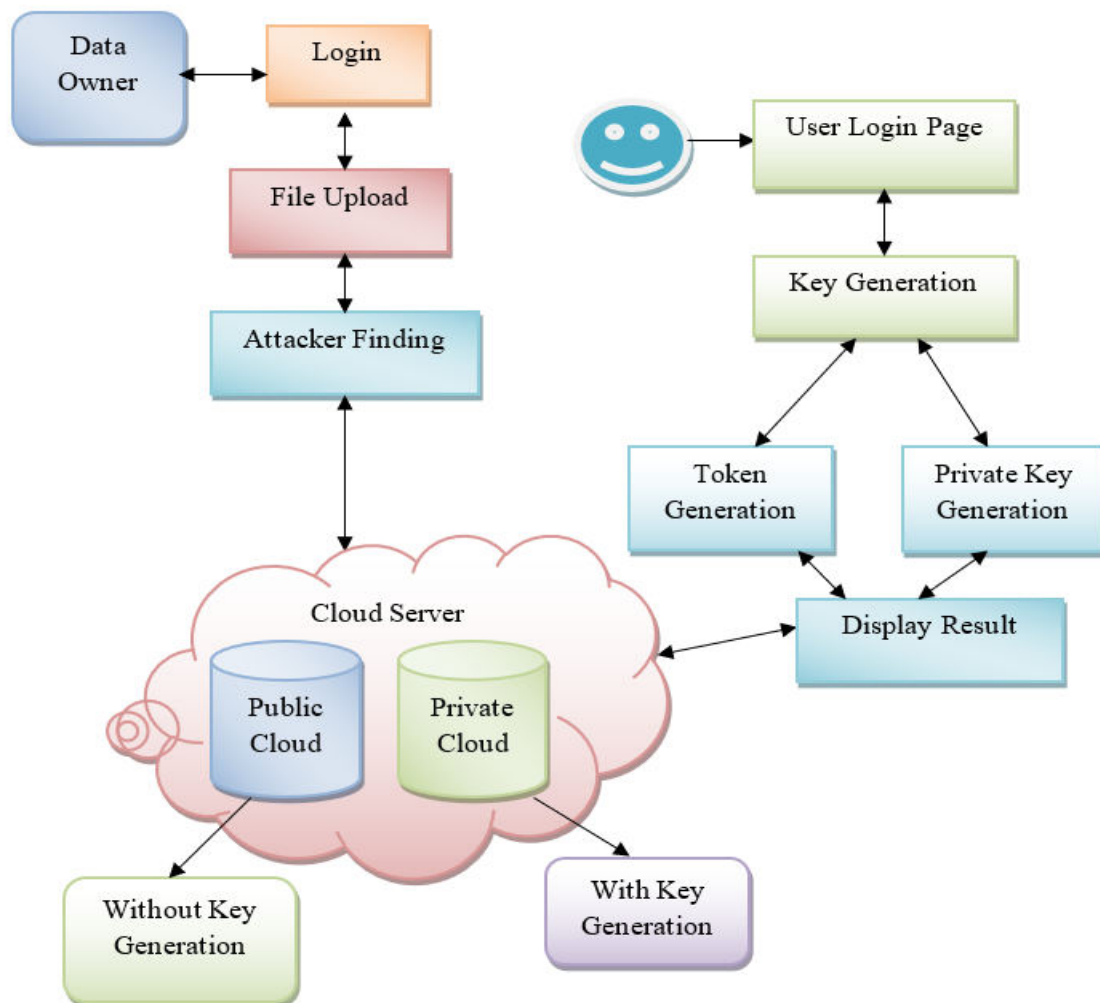


Fig 1.1 System Architecture

V. METHODOLOGY

Modules Name:

- User Interface Design
- Cloud Service Provider
- Authorized Clients
- Key Distribution Center

User Interface Design: To connect to the server, users must provide their username and password. Only then can they access the server. If the user already exists, they can log in directly. Otherwise, new users need to register by submitting details such as username, password, email ID, city, and country. The database will create an account for each user to track their upload and download activity, with the username serving as the user ID. Logging in typically grants access to a specific page where users can search for queries and view.

Cloud Service Provider (CSP): The Cloud Service Provider (CSP) securely stores the encrypted data submitted by clients. It utilizes a pointer-based approach to efficiently manage storage space and mitigate duplicate data. Finally, pointer-based storage at the cloud side eliminates the need to store identical encrypted data blocks by maintaining pointers to existing values, significantly reducing storage requirements.

Authorized Clients (Data Owner): Clients are users who belong to particular groups or organizations and have access to the Key Distribution Center (KDC) to retrieve encryption keys. Before sending any data, clients first obtain the necessary keys from the KDC to encrypt specific data segments, called bases. The data upload process to the Cloud Service Provider (CSP) involves five main steps:

1. The data is broken down into smaller tokens using a tokenization algorithm.
 2. Each token is converted into a base and deviation pair using the Wagner-Fischer algorithm.
 3. A unique identifier for each base is created by calculating its CRC value, which is stored locally for later use.
 4. The base is encrypted with the acquired encryption key and a selected encryption algorithm to ensure confidentiality and integrity.
 5. Finally, the client uploads the encrypted base, its CRC value, and the deviation to the CSP. If the CRC value of a base is already stored locally, only the CRC value and deviation are sent.
- The system is designed to operate within clients limited local storage capacity. Additionally, future enhancements may include integrating advanced cryptographic methods like Verifiable Authenticated Data Structures (VADS) to improve data integrity and auditability.

Key Distribution Center (KDC): The Key Distribution Center (KDC) acts as a central authority that provides encryption keys to authorized clients. When a client needs an encryption key, it sends its unique group ID (IDClient) to the KDC. The KDC then verifies the client's identity and authenticity through a secure authentication method, such as challenge-response or ticketing protocols. Once the client is successfully authenticated, the KDC generates a unique encryption key for the client and securely delivers it to the client's device.

VI. IMPLEMENTATION

Privacy-Preserving Searchable Encryption (PPSE):

This section explores the proposed scheme, DEDUCT, a novel approach to secure and efficient textual data deduplication in cloud storage. It outlines the key steps in the client-side and cloud-side processes.

Client-Side

As mentioned before, the client-side process comprises five steps: Obtaining the Encryption Key, Data Splitting, The core of DEDUCT's deduplication mechanism lies in its combination of tokenization, transformation, CRC computing, and pointer-based storage.

1) Obtaining Encryption Key:

To initiate the data encryption, clients must first obtain the encryption key (k) from the KDC. Since clients are assigned to specific groups or organizations, the KDC must ensure that all authorized users within the same group receive the same encryption key. To verify the client's authenticity, the KDC employs Kerberos authentication, which is a ticket-based authentication mechanism.

2) Tokenization

During this step, the client uses tokenization to divide the data into smaller pieces. Let D be the original data, and $T=t_1, t_2, \dots, t_n$ be the set of tokens resulting from the tokenization algorithm. The tokenization algorithm, denoted as $\text{Tokenize}(D)$, partitions D into a sequence of non-overlapping tokens such that T represents the set of all generated tokens.

3) Transformation:

The transformation step is a critical phase in DEDUCT, involving the conversion of each token into a base and deviation, as defined below: Let D denote the original data, and $T=t_1, t_2, \dots, t_n$ represent the set of tokens resulting from the tokenization algorithm. The Transformation process, denoted as $\text{Transform}(t_i)$, converts the input token into a corresponding base-deviation pair (b_i, d_i) .

VII. EXPERIMENTAL RESULTS

Home Page:



Fig 2: Home Page

The home page which includes CSP, USER, Key Distribution Center and SIGN UP. These are the main navigation tags which are important in the whole project execution. All user and cloud service provider operations are performed here.

CSP Page:

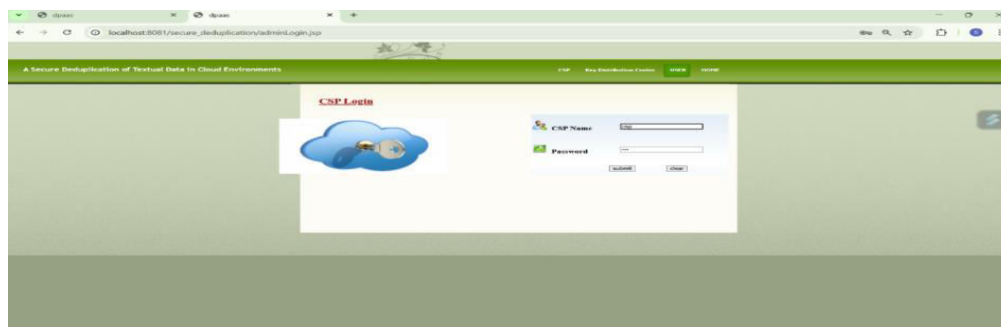


Fig 3: CSP Page

CSP stands for Cloud Service Provider. It is a cloud side container where the user access should be done and no permission given without csp login. Cloud Service Provider ensures the risk of side-channel attacks, where unauthorized access to files uploaded by other user.

User Login Page:

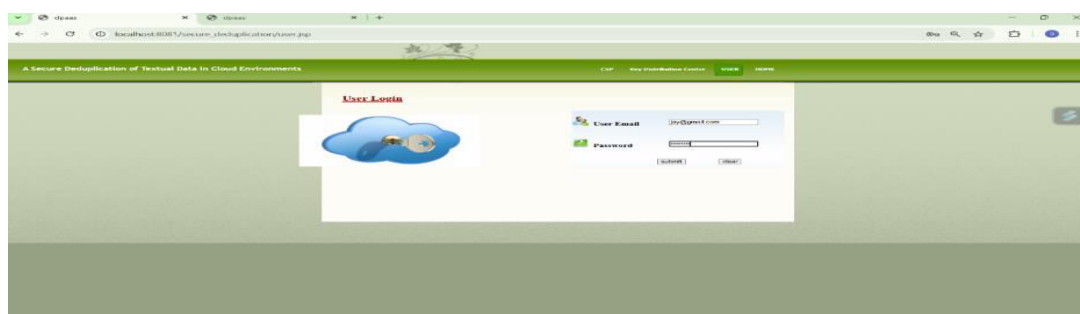


Fig 4: User Login Page

After registering CSP logins and access and checks the user registration form if it is official and not harmful. Then you can go to user tab and enter user email and password to login.

User Operations:



Fig 5: User Operations

After User login. It can show above interface which user can perform various operations such as Data production, Data Details, File Token, File Download and Duplicate File Download.

Private Cloud:

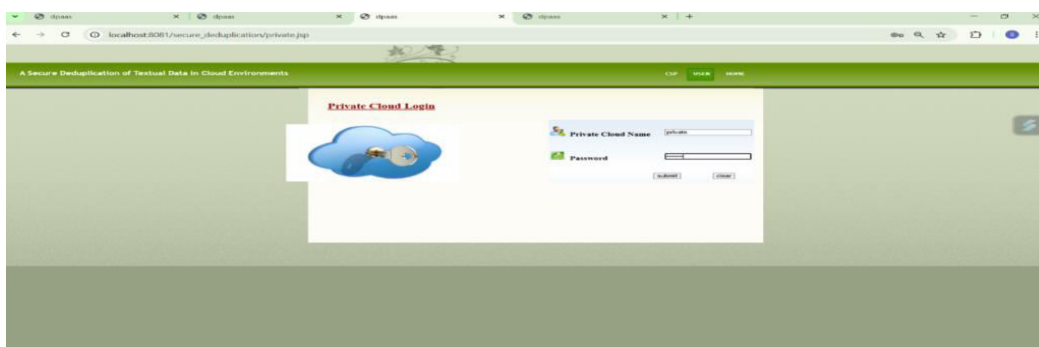


Fig 6: Private Cloud

Private Cloud is only accessed by the cloud service provider to enhance security while uploading user data. This private cloud is just for only storing data temporarily and enhance data confidentiality.

Public Cloud:

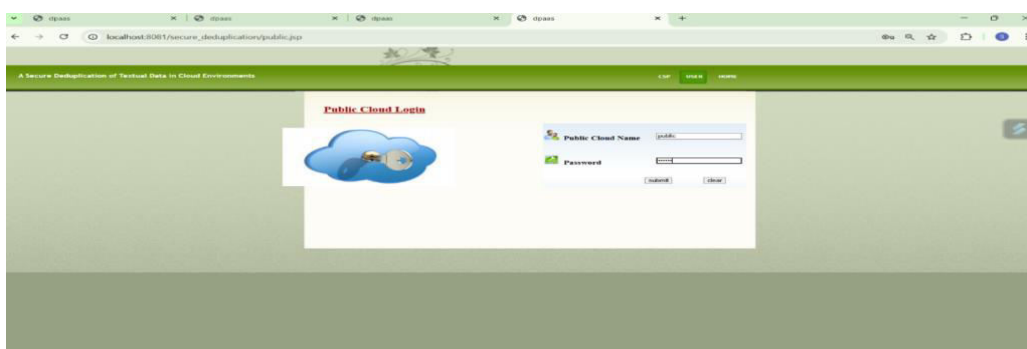
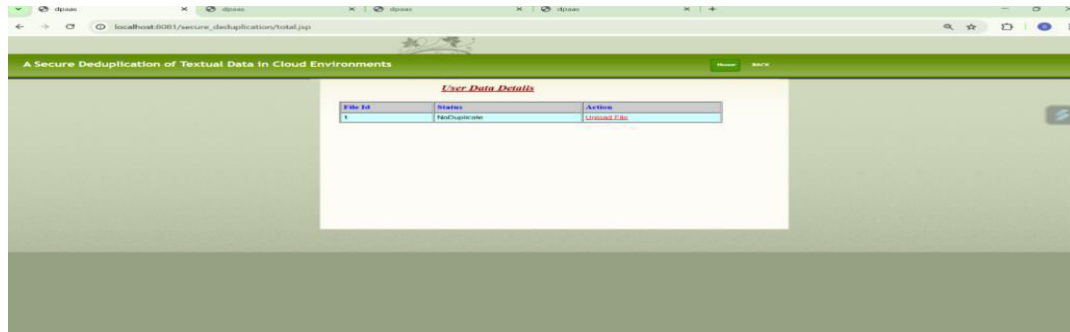


Fig 7: Public Cloud

Public cloud is used to store user data permanently and it accepts the permission from the user file data operations. After user login the request sent directly to public cloud to upload the file.

Data Details:



File ID	Status	Action
1	NotDuplicate	Upload File

Fig 8: Data Details

It contains user data details which includes the uploaded files of the user. The data is whether duplicate or no duplicate, it will display on the status field of the User Data Details Tab.

Request File:



File ID	Status	Action
1	NotDuplicate	Upload File
2	NotDuplicate	Upload File
3	Duplicate	Download the Uploaded

Fig 9: Request File

Now if user uploads another file with same content then it shows as duplicate file. If we want to view, then we can request the file. Then it can generate the hash value on MySQL where we can copy and paste then downloading the file.

VIII. CONCLUSION

This project presents DEDUCT, a textual deduplication technique that leverages generalized deduplication and client-side pre-processing to significantly enhance cloud storage efficiency and data security. DEDUCT demonstrates notable improvements in these key areas compared to existing state-of-the-art methods. DEDUCT achieves a compression ratio of 66% which translates to direct cost savings and improved scalability for cloud storage solutions, offering increased capacity and reduced financial burden. Moreover, DEDUCT's design is well-suited for resource-constrained devices commonly found. This adaptability addresses crucial needs in resource-limited environments where efficient data handling is critical. While the evaluation focused on the Touchdown dataset, DEDUCT's applicability extends to broader domains. Its strengths in efficiently deduplications large textual datasets make it highly relevant to mobile, and embedded systems, where storage and bandwidth are often limited. DEDUCT's flexibility and resource-friendly approach offer valuable solutions for these areas.

IX. FUTURE ENHANCEMENT

We aim to enhance client-side pre-processing techniques for future work by utilizing natural language processing and machine learning algorithms. Employing advanced tokenization and lemmatization algorithms can enhance the accuracy of near-duplicate data identification while reducing computational overhead. Additionally, energy efficiency is paramount in resource-constrained environments like edge computing devices, and DEDUCT can be further optimized to address this concern.

REFERENCES

- [1] P. Anderson, Q. Wu, D. Teney, J. Bruce, M. Johnson, N. Sünderhauf, I. Reid, S. Gould, and A. van den Hengel, "Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., Jun. 2018, pp. 3674–3683, doi: 10.1109/CVPR.2018.00387.
- [2] W. Xia, H. Jiang, D. Feng, F. Douglass, P. Shilane, Y. Hua, M. Fu, Y. Zhang, and Y. Zhou, "A comprehensive study of the past, present, and future of data deduplication," Proc. IEEE, vol. 104, no. 9, pp. 1681–1710,
- [3] P. Prajapati and P. Shah, "A review on secure data deduplication: Cloud storage security issue," J. King Saud Univ. Comput. Inf. Sci., vol. 34, no. 7, pp. 3996–4007, Jul. 2022, doi: 10.1016/j.jksuci.2020.10.021.
- [4] D. T. Meyer and W. J. Bolosky, "A study of practical deduplication," ACM Trans. Storage, vol. 7, no. 4, pp. 1–20, Jan. 2012, doi: 10.1145/2078861.2078864.
- [5] OpenDedup. (2023). OpenDedUp. Accessed: Aug. 6, 2023. [Online]. Available: <http://opendedup.org/>
- [6] S. Keelveedhi, M. Bellare, and T. Ristenpart, "DupLESS: Server-Aided encryption for deduplicated storage," in Proc. 22nd USENIX Secur. Symp. (USENIX Secur.), 2013, pp. 179–194.
- [7] J. Liu, N. Asokan, and B. Pinkas, "Secure deduplication of encrypted data without additional independent servers," in Proc. ACM SIGSAC Conf., Oct. 2015, pp. 874–885, doi: 10.1145/2810103.2813623.
- [8] K. Ghassabi, P. Pahlevani, and D. E. Lucani, "Deduplication of textual data by NLP approaches," in Proc. IEEE 97th Veh. Technol. Conf. (VTC-Spring), Florence, Italy, Jun. 2023, pp. 1–6, doi: 10.1109/vtc2023-spring57618.2023.10199538.
- [9] K. Jin and E. L. Miller, "The effectiveness of deduplication on virtual machine disk images," in Proc. Israeli Exp. Syst. Conf., May 2009, pp. 1–12, doi: 10.1145/1534530.1534540.
- [10] S. Lee and D. Choi, "Privacy-preserving cross-user source-based data deduplication in cloud storage," in Proc. Int. Conf. ICT Converg. (ICTC), Oct. 2012, pp. 329–330, doi: 10.1109/ICTC.2012.6386851.
- [11] B. Wang, W. Lou, and Y. T. Hou, "Modeling the side-channel attacks in data deduplication with game theory," in Proc. IEEE Conf. Commun. Netw. Secur. (CNS), Sep. 2015, pp. 200–208, doi: 10.1109/CNS.2015.7346829.
- [12] F. Armknecht, C. Boyd, G. T. Davies, K. Gjosteen, and M. Toorani, "Side channels in deduplication," in Proc. ACM Asia Conf. Comput. Commun. Secur., Apr. 2017, pp. 266–274, doi: 10.1145/3052973.3053019.
- [13] H. Chen, A. Suhr, D. Misra, N. Snively, and Y. Artzi, "TOUCHDOWN: Natural language navigation and spatial reasoning in visual street environments," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 12530–12539, doi: 10.1109/CVPR.2019.01282.
- [14] R. Vestergaard, Q. Zhang, and D. E. Lucani, "Generalized deduplication: Bounds, convergence, and asymptotic properties," in Proc. IEEE Global Commun. Conf. (GLOBECOM), Dec. 2019, pp. 1–6, doi: 10.1109/GLOBECOM38437.2019.9014012.
- [15] H. Sehat, E. Pagnin, and D. E. Lucani, "Yggdrasil: Privacy-aware dual deduplication in multi-client settings," in Proc. IEEE Int. Conf. Commun., Jun. 2021, pp. 1–6, doi: 10.1109/ICC42927.2021.9500816.
- [16] L. Nielsen and D. E. Lucani, "Hekate a tool for gauging data deduplication performance," in Proc. IEEE 6th Int. Conf. Smart Cloud (SmartCloud), Nov. 2021, pp. 67–72, doi: 10.1109/SmartCloud52277.2021.00019.
- [17] Z. Pooranian, K.-C. Chen, C.-M. Yu, and M. Conti, "RARE: Defeating side channels based on data-deduplication in cloud storage," in Proc. IEEE Conf. Comput. Commun. Workshops (INFOCOM WKSHPS), Apr. 2018, pp. 444–449, doi: 10.1109/INFCOMW.2018.8406888.
- [18] S. Zhang, H. Xian, Z. Li, and L. Wang, "SecDedup: Secure encrypted data deduplication with dynamic ownership updating," IEEE Access, vol. 8, pp. 186323–186334, 2020, doi: 10.1109/ACCESS.2020.3023387.
- [19] K. Akhila, A. Ganesh, and C. Sunitha, "A study on deduplication techniques over encrypted data," Proc. Comput. Sci., vol. 87, pp. 38–43, Jan. 2016, doi: 10.1016/j.procs.2016.05.123.
- [20] C.-M. Yu, S. P. Gochhayat, M. Conti, and C.-S. Lu, "Privacy aware data deduplication for side channel in cloud storage," IEEE Trans. Cloud Comput., vol. 8, no. 2, pp. 597–609, Apr. 2020, doi: 10.1109/TCC.2018.2794542.

International Journal of Advanced Research in Education and Technology

ISSN: 2394-2975

Impact Factor: 8.152