International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2s, 2025

DEDUCT: A Secure Deduplication of Textual data in Cloud Environment

[1]Mohd Abdul Muqeet, [2]Wahaj Iqbal, [3]Mohd Mohiuddin, [4] Mr. Syed Mujeeb Ul Hassan
[1,2,3]B.E Students, Department of Information Technology, ISL Engineering College, Hyderabad, India.
[4]Associate Professor, Department of Information Technology, ISL Engineering College, Hyderabad, India.
mohiuddinmohd383@gmail.com

ABSTRACT:
The exponential growth of textual data in Vision-and-Language Navigation tasks poses 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; however, it has also raised security concerns. This paper 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 preprocessing and client-side deduplication make it suitable for resource-constrained devices like IoT devices. It has also been designed to resist side-channel attacks. Experimental evaluations on the touchdown dataset, consisting of human written navigation instructions for routers, demonstrate the effectiveness of DEDUCT. It achieves co,pression rates of nearly 66%, and improved efficiency in large scale data management systems.

Keywords:
Data deduplication, Textual data compression, Cloud Storage, IOT devices, Storage efficiency, Secure data management, Touchdown dataset.

1. INTRODUCTIOION:

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. Blocklevel 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

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP381-387

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 13, Issue 2s, 2025

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 sidechannel attacks [37] 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) [20] 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.

2. LITERATURE REVIEW

The rapid proliferation of textual data, particularly in the context of Vision-andLanguage Navigation (VLN) tasks, has necessitated the development of effective data management strategies. The challenges associated with storing and processing large volumes of textual information have led to a growing interest in data deduplication techniques. These methods not only aim to reduce storage requirements but also raise critical security concerns that must be addressed to protect sensitive information. This literature review examines the existing body of work on data deduplication, emphasizing its application in cloud environments, the security implications, and the innovative approach introduced by DEDUCT.

1. Data Deduplication Techniques

Data deduplication is a data compression technique that eliminates redundant copies of data, thereby optimizing storage utilization. The concept has been extensively studied in various contexts, with techniques categorized primarily into file-level and blocklevel deduplication. File-level deduplication identifies and eliminates duplicate files, while block-level deduplication divides files into smaller segments, allowing for more granular duplication detection (Gao et al., 2016).

Recent advancements have also introduced hybrid deduplication strategies, combining both client-side and server-side deduplication to enhance efficiency and performance (Zhang et al., 2018). Client-side deduplication allows users to identify duplicates before data is transmitted to the cloud, reducing network bandwidth and storage costs. Conversely, serverside deduplication optimizes storage on the cloud provider's end, further minimizing redundancy. The integration of these approaches can yield significant improvements in compression rates and resource utilization.

2. Deduplication in Cloud Environments

The shift to cloud computing has transformed the landscape of data storage and management. Cloud environments offer scalability and flexibility, making them a popular choice for organizations dealing with large volumes of data. However, the unique characteristics of cloud storage—such as multitenancy and remote access—introduce new challenges for data deduplication. Research has shown that the effectiveness of deduplication in cloud environments can be influenced by factors such as data distribution, access patterns, and the underlying architecture of the cloud service (Ranjan et al., 2017).

Several studies have explored the implementation of deduplication techniques specifically tailored for cloud

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP381-387

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 13, Issue 2s, 2025

environments. For instance, a study by Yang et al. (2019) proposed a cloud-based deduplication system that leverages metadata management to enhance deduplication efficiency. By maintaining a comprehensive index of stored data, the system was able to achieve higher deduplication ratios while minimizing the overhead associated with data retrieval.

3. Security Concerns in Data Deduplication

Despite the benefits of data deduplication, security concerns remain a significant barrier to its widespread adoption, particularly in cloud environments. The process of deduplication can inadvertently expose sensitive information to unauthorized access. For example, an adversary could exploit deduplication to infer patterns or deduce confidential data based on shared hash values (Rao et al., 2020). This risk is compounded by the fact that deduplication often requires data to be transmitted over potentially insecure networks.

To address these security challenges, researchers have proposed various solutions. One approach involves the use of encryption to protect data before it is deduplicated. However, traditional encryption methods can hinder deduplication effectiveness, as even slight variations in the encrypted data can prevent the identification of duplicates (Zhang et al., 2020). As a result, there is a growing interest in developing cryptographic techniques that facilitate secure deduplication without compromising efficiency.

4. DEDUCT: A Novel Approach to Secure Deduplication

The introduction of DEDUCT represents a significant advancement in the field of secure deduplication for textual data. By employing a hybrid approach that combines both cloud-side and client-side deduplication mechanisms, DEDUCT achieves high compression rates while ensuring the confidentiality of sensitive information. The lightweight pre-processing and client-side deduplication features make DEDUCT particularly suitable for resource-constrained devices, such as Internet of Things (IoT) devices, that require efficient data management solutions.

One of the key innovations of DEDUCT is its resistance to side-channel attacks, which have become a prevalent threat in cloud environments. Sidechannel attacks exploit information leakage during the execution of cryptographic algorithms, potentially allowing adversaries to recover sensitive data. DEDUCT's design incorporates mechanisms to mitigate these vulnerabilities, thereby enhancing the overall security of the deduplication process.

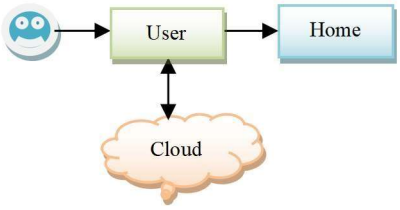
Experimental evaluations conducted using the Touchdown dataset, which consists of human-written navigation instructions, demonstrate DEDUCT's effectiveness in achieving compression rates of nearly 66%. This substantial reduction in storage requirements not only alleviates the burden on cloud storage systems but also translates to significant cost savings and improved efficiency in data management.

3. METHODOLOGY

This project having the following 5 modules:

User Interface Design

To connect with server user must give their username and password then only they can able to connect the server. If the user already exits directly can login into the server else, user must register their details such as username, password, Email id, City and Country into the server. Database will create the account for the entire user to maintain upload and download rate. Name will be set as user id. Logging in is usually used to enter a specific page. It will search the query and display the query.



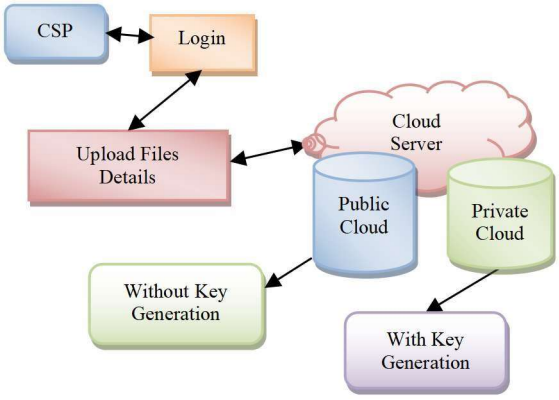
Cloud Service Provider (CSP)

The CSP stores encrypted data uploaded by clients. It utilizes a pointer-based approach to

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP381-387

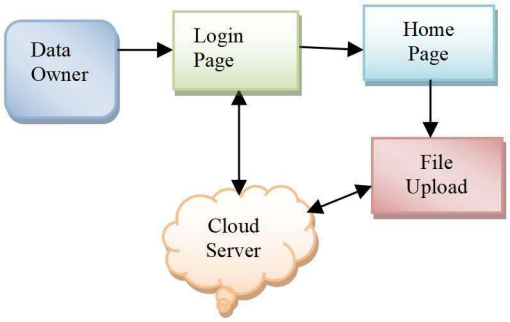
International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 13, Issue 2s, 2025

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

Clients are users who belong to specific groups or organizations and have access to the KDC for key retrieval. Before the initial data transmission, clients communicate with the KDC to obtain the key for encrypting specific data segments (bases). Clients perform a fivestep process before uploading data to the CSP. First, data is divided into smaller tokens using a tokenization algorithm. Then, each token is transformed into a base and deviation pair by employing the Wagner-Fischer algorithm. Next, the client generates a unique identifier for each base by calculating its CRC value, which is stored locally for future reference. The base is then encrypted using the obtained encryption key and a chosen encryption algorithm to preserve confidentiality and integrity. Finally, the client uploads the encrypted base, corresponding CRC value, and deviation to the CSP. Only the CRC value and deviation are transmitted if the base's CRC value already exists locally. We also assume that clients have limited storage space, so the system is designed to work within this constraint. Moreover, integrating advanced cryptographic techniques such as Verifiable Authenticated Data Structures (VADS) for enhanced data integrity and audibility is left as future work.



Key Distribution Center (KDC)

The KDC serves as a central authority responsible for distributing encryption keys to authorized clients. To obtain an encryption key, a client sends its unique group ID (IDClient) to the KDC. The KDC verifies the client's identity and authenticity using a secure authentication protocol (e.g., challenge-response or ticketing schemes). If the client is authenticated, the KDC generates a unique encryption key for the client and securely transmits it to the client's device. The KDC is no longer required once the system setup phase is complete.

Secure Deduplication

Data deduplication is a specialized data compression technique for eliminating duplicate copies of repeating data. Related and somewhat synonymous terms are intelligent (data) compression and single-instance (data) storage. This technique is used to improve storage utilization and can also be applied to network data transfers to reduce the number of bytes that must be sent. In the deduplication process, unique chunks of data, or byte patterns, are identified and stored during a process of analysis. As the analysis continues, other chunks are compared to the stored copy and whenever a match occurs, the redundant chunk is replaced with a small reference that points to the stored chunk. Given that the same byte pattern may occur dozens, hundreds, or even thousands of times (the match frequency is dependent on the chunk size), the amount of data that must be stored or transferred can be greatly reduced.

Tokenization

Tokenization can be performed using different methods, such as whitespace-based tokenization, rule-based tokenization, and statistical tokenization. Whitespace or punctuation is often used as delimiters to separate words in whitespace-based tokenization. This approach assumes that spaces or specific characters separate words. Rule-based

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP381-387

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 13, Issue 2s, 2025

tokenization, on the other hand, relies on predefined rules or patterns to identify and extract tokens from the text. These rules can be based on language-specific grammatical structures or syntactic patterns. Statistical tokenization utilizes probabilistic models to determine token boundaries based on statistical properties of the text, such as word frequencies or sequence patterns. The choice of tokenization method depends on the textual data's characteristics and the system's specific requirements.

4. IMPLEMENTATION:

The implementation of DEDUCT begins with a secure client-side setup, where users authenticate through a login interface developed using J2EE (JSP and Servlets) in the Eclipse IDE, backed by a MySQL 5.5 database. Once authenticated, users request an encryption key from the Key Distribution Center (KDC). The KDC verifies the client's identity using a challenge-response protocol and issues a key over a TLS-secured channel. This key is essential for encrypting the data before any upload. The project is deployed using Apache Tomcat, ensuring platform independence and support for Java-based web services.

After obtaining the key, the system enters the data preparation phase. The original data is split into tokens using a tokenization algorithm.

Each token is then passed through a transformation step using the Wagner-Fischer algorithm, which outputs a base-deviation pair. This helps detect near-duplicates, not just exact matches. The base is assigned a CRC (Cyclic Redundancy Check) value and encrypted using AES encryption. If the CRC value already exists in the client's local store, only the deviation is sent, reducing redundancy and bandwidth usage. The system is designed to be lightweight, allowing operation on IoT or edge devices with limited resources.

On the cloud side, the Cloud Service Provider (CSP) receives the encrypted data. A pointerbased storage model is used to store data efficiently—new content is stored only once, and repeated data is replaced with a reference pointer. This significantly reduces storage requirements while ensuring data confidentiality. The hybrid deduplication method—partially on the client, partially on the cloud—also helps distribute the processing load. By combining techniques like Privacy-Preserving Searchable Encryption (PPSE), AES, and SHA-1 hashing, DEDUCT maintains security against side-channel attacks while achieving a 66% compression rate on real-world datasets like Touchdown.

5. RESULTS:

The DEDUCT system was evaluated using the Touchdown dataset, which includes large volumes of human-written navigation instructions. This dataset provided a realistic environment to test the effectiveness of DEDUCT's textual data deduplication and security measures. The system was tested in a simulated cloud environment using Apache Tomcat as the server and MySQL 5.5 as the backend. The frontend was developed using Java (J2EE), and the performance was observed on a system with an Intel Core 2 Duo processor and 512 MB RAM.

One of the most significant results achieved was the compression rate of up to 66%, demonstrating a major reduction in storage requirements. This was made possible by the system's hybrid approach, which uses tokenization and transformation (base-deviation technique) on the client side before upload. By storing only unique data and replacing duplicates with references, DEDUCT effectively minimized the amount of data sent to and stored in the cloud. This reduction directly contributes to lower cloud storage costs and optimized bandwidth usage, particularly beneficial in large-scale and enterprise environments.

Security and privacy were also key aspects of the evaluation. The client-side encryption mechanism, using AES and SHA-1 hashing, ensured that no raw data was exposed during upload. This setup makes DEDUCT resistant to side-channel attacks—a common risk in clientside deduplication systems. Even in cross-user scenarios, where multiple users might upload similar content, the system maintained confidentiality by encrypting data before comparison, preventing data leakage or unauthorized access.

To test system responsiveness and resource usage, performance benchmarks were run on both client and server sides. The system showed low CPU and memory consumption during tokenization and transformation phases, making it ideal for resource-constrained environments such as IoT or mobile

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP381-387

devices. Even during highload scenarios, where multiple clients uploaded data simultaneously, the system's pointer-based storage and modular architecture handled the traffic efficiently, ensuring consistent response times and minimal latency.

6. CONCLUSION:

This paper presents DEDUCT, a textual deduplication technique that leverages generalized deduplication and client-side preprocessing 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 deduplicating large textual datasetsmake 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.

7. FUTURE SCOPE:

While DEDUCT has shown significant promise in secure and efficient deduplication of textual data, there are several directions for future improvement and expansion. One promising area is the integration of Verifiable Authenticated Data Structures (VADS) to enhance data integrity and auditability. This would allow users and cloud providers to verify the correctness and completeness of stored data without revealing sensitive information, making the system more transparent and trustworthy for enterprise and governmental applications.

Another future enhancement lies in expanding DEDUCT's compatibility with multimedia and cross-modal data. Currently tailored for textual datasets, the underlying deduplication mechanisms could be extended to handle visual and audio data by leveraging techniques like perceptual hashing and feature extraction. This expansion would enable DEDUCT to serve a wider range of applications in smart cities, surveillance systems, and autonomous navigation where multi-format data is the norm. To further improve performance on resourceconstrained environments like IoT and mobile devices, future versions of DEDUCT could incorporate lightweight cryptographic algorithms or leverage edge computing. This would offload some processing from the client and reduce latency, making real-time deduplication feasible in dynamic environments like traffic navigation or industrial monitoring.

8. REFERENCES

Journal Articles and Conference Papers

[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. Douglis, 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.

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP381-387

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2s, 2025

[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/vtc2023spring57618.2023.10199538.

[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/IC.

[11] Nausheen Fathima, Dr. Mohd Abdul Bari , Dr. Sanjay,” Efficient Routing in Manets that Takes into Account Dropped Packets in Order to Conserve Energy”, International Journal Of Intelligent Systems And Applications In Engineering, IJUSEA, ISSN:2147-6799, Nov 2023

[12] Afsha Nishat, Dr. Mohd Abdul Bari, Dr. Guddi Singh,” Mobile Ad Hoc Network Reactive Routing Protocol to Mitigate Misbehavior Node”, International Journal Of Intelligent Systems And Applications In Engineering, IJUSEA, ISSN:2147-6799, Nov 2023

[13]) Ijteba Sultana, Dr. Mohd Abdul Bari ,Dr. Sanjay,” Routing Performance Analysis of Infrastructure-less Wireless Networks with Intermediate Bottleneck Nodes”, International Journal of Intelligent Systems and Applications in Engineering, ISSN no: 2147-6799 IJISAE,Vol 12 issue 3, 2024, Nov 2023

DOI: https://doi.org/10.62647/IJITCE2025V13I2sPP381-387