Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.98549881

EFFECTIVE DATA REDUCTION TECHNIQUES FOR INTERNET OF THINGS (IOT)

Nilima Dongre Jawade, Zeba Shaikh, Dr. Atul D. Raut, Dr. Mohammad Atique

Department of Computer Science & Engineering, Sant Gadge Baba Amravati University, Amravati,

Correspondence should be addressed to Nilima Dongre Jawade; neilimarj2@gmail.com

Abstract

The Internet of Things (IoT) has witnessed a rapid growth in the number of connected devices, leading to an exponential increase in data generated by these devices. This data explosion poses significant challenges in terms of storage, processing, and communication in IoT systems. To address these challenges, effective data reduction techniques have emerged as a promising solution. Data reduction techniques aim to reduce the volume, complexity, and redundancy of IoT data while preserving its key information and minimizing the impact on application quality. Data reduction techniques play a crucial role in managing the ever-growing volumes of data generated in various domains. Deduplication and compression are two popular approaches employed for data reduction. Deduplication eliminates redundant data by identifying and removing duplicate instances, while compression reduces data size by encoding it in a more compact form. It is mostly used in cloud to reduce the amount of storage space in the cloud. Among the data reduction techniques used are compression, data deletion, archiving and data deduplication. De-duplication supports the confidentiality of sensitive data. We propose a system based on client-side de-duplication technique which efficiently and effectively saves the storage space. The proposed system EDAA (Elimination, Dynamic Ownership, Authentication, Authorization) encrypts the data before uploading it to the cloud storage. EDAA supports de-duplication along with dynamic ownership to authorized user. We show that our proposed EDAA system sustain very little overhead as compared to normal operations.

1. INTRODUCTION

It is widely acknowledged that the current society has entered the era of big data, and there is a massive surge in various types of data. Based on research reports, IDC estimates that the storage capacity of the global market will grow exponentially from 33 ZB to 173 ZB between 2018 and 2025[1].

Cloud storage is a cloud computing model also known as utility storage in which data is maintained, managed, backed up remotely and made available to users over network. Duplication happens when multiple users upload the same data to the cloud storage. De-duplication is a technique used to eliminate duplicate copies of identical data and it is used in cloud storage to save bandwidth and reduce storage space. Now-a-days, a lot of data is being uploaded on the cloud storage. Mostly similar data is outsourced by different users on the same cloud platform. Hence, there is a necessity to detect these duplicate data and further eliminate

these duplicate data by keeping the original set of data. This saves the storage space used by the duplicate data.

There are many data reduction techniques and various approaches, including compression, aggregation, summarization, sampling, and filtering. These techniques leverage statistical analysis, data mining, machine learning, and other computational methods to extract relevant information from IoT data streams while minimizing storage requirements, energy consumption, and network bandwidth utilization. This [2] paper provides a comprehensive survey of data reduction techniques specifically designed for the Internet of Things (IoT). It explores various approaches to reduce data size and complexity in IoT systems, including compression, aggregation, sampling, and filtering. The paper discusses the benefits, challenges, and limitations of these techniques and provides insights into their applications in different IoT scenarios. The review paper [3] focuses on data reduction techniques in the context of the IoT. It provides an overview of different methods and algorithms for reducing data size, energy consumption, and network bandwidth utilization in IoT deployments. The paper discusses the advantages and limitations of each technique and highlights their implications for IoT applications and systems. This paper [4] presents a review of data reduction techniques specifically tailored for IoT sensor data. It discusses various methods to reduce sensor data volume, including compression, summarization, and feature extraction techniques. The paper examines the trade-offs between data reduction and information loss and provides insights into the applicability of these techniques in IoT sensor networks. This survey paper [5] focuses on data reduction techniques for wireless sensor networks (WSNs) within the IoT context. It provides an overview of different data reduction approaches, such as spatial and temporal correlation-based techniques, data aggregation, and in-network processing. The paper discusses the benefits and challenges of these techniques in WSNs and their relevance to IoT applications. The survey paper [6] presents a comprehensive overview of data reduction techniques for efficient IoT data management. It explores various strategies to reduce data size, including compression, deduplication, and data summarization techniques. The paper discusses the impact of data reduction on energy consumption, storage requirements, and data processing in IoT systems, and highlights their potential benefits and challenges.

Most significant data reduction techniques are compression, deduplication, encryption, and data deletion. More ways of reducing the data, is by archiving it and using a tiered storage. In our methodology we are utilizing compression and deduplication. The duplicate data is simply replaced by a pointer, we propose an algorithm to reduce the data using compression and deduplication. Firstly, deduplication is examined as an effective method to eliminate duplicate data. Various deduplication approaches, such as content-based chunking, fixed-size and variable-size chunking, and fingerprint-based methods, are explored. The advantages and limitations of each technique are discussed, along with their application scenarios. Secondly, compression techniques are investigated as a means to reduce the size of data. Different compression algorithms, including lossless and lossy compression, are reviewed. The focus is on popular compression algorithms such as gzip, Lempel-Ziv-Welch (LZW), and Burrows-Wheeler Transform (BWT). The trade-offs between compression ratios, computational

complexity, and decompression overhead are examined. Furthermore, the synergy between deduplication and compression is explored. Deduplication is capable of identifying redundancy across data segments, while compression algorithms can exploit the inherent redundancy within each segment. The combination of these techniques enables enhanced data reduction by eliminating both intra-segment and inter-segment redundancy. The performance implications of data reduction using deduplication and compression are also discussed. Factors such as data access patterns, system architecture, and computational overhead are considered. Additionally, the impact of deduplication and compression on data integrity, recovery, and system throughput are examined.

2. SURVEY OF DIFFERENT METHODOLOGY

Duplication happens when multiple copies of same data and that to at multiple locations. For data reduction the first technology comes into picture is compression. String matching is a fundamental component of compression technology used to identify and retrieve identical data. It relies on string-matching algorithms and their variations, which excel at accurately identifying matches. Although the implementation of precise matching can be intricate, it offers a high level of accuracy and effectively eliminates fine-grained redundancy.

Data deduplication

Data de-duplication is a technique used to eliminate duplicate copies of identical data, it may be file or block, and it is used to save bandwidth and reduce storage space. Being in big data era, a lot of data is being produced and stored or uploaded on the local devices, edge or network devices and cloud devices. Hence, there is a necessity to detect these duplicate data and further eliminate these duplicate data by keeping the original set of data. This saves the storage space used by the duplicate data. The duplicate data is simply replaced by a pointer.

Based on granularity there are two types of de-duplication viz i) File-level de-duplication, ii) Block-level de-duplication

File-level de-duplication:

File-level de-duplication compares entire file based on hash value of the file further eliminating the duplicate file. Suppose we have two identical files A and B and upload the file A. This technique computes the hash value of file A and if the hash value is not found in the database, file A is unique. Later if file B is uploaded and its hash value is found in database, so the corresponding file is not saved.

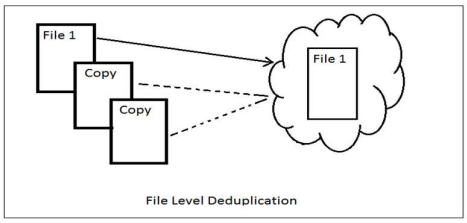

Fig 1: File Level De-duplication

Block-level de-duplication:

Block-level de-duplication divides the entire file either in fixed number of blocks or fixed sized

blocks. It computes hash value of each block and compares it with the hash value of previously uploaded blocks of the file.This technique provides better de-duplication ratio as compared to the File-level de-duplication.

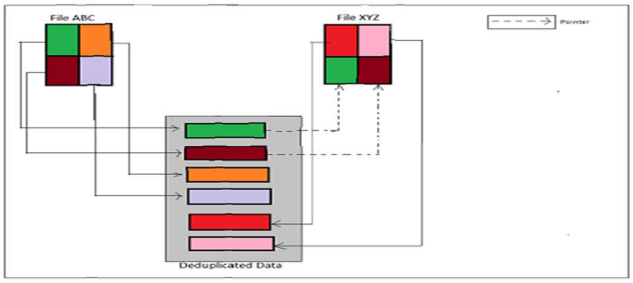

Fig 2: Block-level De-duplication

1. Related Work:

The authors [7] did the performance comparative study for de-duplication techniques. They concluded that chunk level de-duplication achieves better efficiency because it does exact de-duplication throughput is less since it checks every incoming chunks. However, file level de-duplication is less efficient because some duplicate blocks of data may be present across different groups. The authors [8] reviewed various de-duplication techniques. They conclude that fixed size chunking method solve file level chunking method but fails if few bytes are added or removed from the data. Hence, byte sniffing problem occurs. The variable size blocks solve the problem creating boundaries based on content inside the files. Content level chunking method provides good throughput as well as decrease in space utilization. The authors [9] presented hybrid cloud approach for secure authorized de-duplication using block level de-duplication technique. It aims for solving problems of de-duplication with features such as authorized duplicate check, unforgettability of file token, data confidentiality. However, as the block size number of block increases the mapping time increases. The authors [10] fixed the clauses of unaligned data by using variable length de-duplication. They implemented identifier of duplicate data using fixed size block and variable size block de-duplication over real world dataset. They concluded that variable size saves more space than fixed size. However, handling of data is more complex in variable size block than fixed size. The authors [11] reviewed various deduplication techniques in cloud storage. They concluded that Block level deduplication ad more accuracy than file level deduplication. They proposed algorithm that can find out duplication of data in the uploaded file before storage which supports for both file and block level deduplication. The authors [12] reviewed various techniques and approaches used for deduplication over encrypted data. The authors [13] proposed server-side deduplication for encrypted file also an efficient group key management protocol ‘Elliptic Curve Cryptography’ in distributed group communication. The authors [14] described comprehensive structure of deduplication compression systems that incorporate both duplicate and resemblance detection. The authors took a detailed insights into delta compression techniques and algorithms. The compression is used on the chunks originally deduced from the data.

3. TABLE 1. COMPARATIVE ANALYSIS

Citation	Encrypted deduplication	Ownership Management	Techniques	Algorithms	Results
[7]	Yes	No	File and Block Level	MD5	50% of Storage is saved
[8]	Yes	-	Comparison between Fixed and Variable sized Blocks	-	Variable Block technique gives best possible result
[9]	Yes	Yes	Block Level	Convergent Encryption Technique	Block Level is better than File level
[10]	Yes	-	Variable Sized Block	-	Variable Block saves 15% more space than Fixed Block
[11]	Yes	No	File and Block Level	MD5	No duplicate file saved in cloud which helps to save storage space.
[12]	Yes	Yes	File Level	Convergent Encryption Technique	System enhances data privacy and confidentiality and fine-grained ownership management
Proposed System (EDAA)	Yes	Yes	File, Block and Content Level	AES SHA-256	Dynamic data ownership, both file-level and block level

The authors [15] reviewed that deduplication offers several advantages in terms of optimizing storage space and exploring diverse deduplication techniques across various dimensions for data management. They concluded by employing deduplication, organizations can effectively maximize their storage utilization and minimize memory wastage caused due to redundant data. Additionally, they also studied different types of deduplication techniques to enable the selection of the most suitable approach to enhance overall storage efficiency while reducing the storage footprint attributed to duplicated data. The authors [16] proposed the DARE scheme, also known as Duplicate-Adjacency based Resemblance Detection (DupAdj), incorporates an

advanced super-feature approach to improve the efficiency of resemblance detection. By examining duplicate data chunks, their system effectively identifies redundancy and achieves higher throughput. Compared to traditional super-feature approaches, DARE demonstrates reduced computation and indexing overheads, approximately 1/4 and 1/2 respectively. Furthermore, the system leverages existing duplicate-adjacency information to strike a unique balance, enabling the identification of the optimal "sweet spot" for the super-feature approach. In both real and synthetic scenarios, the DARE system exhibits 2-10 percent greater redundancy detection, resulting in enhanced overall performance. The authors [17] introduced a secure data deduplication method that incorporated with an efficient Proof-of-Work (PoW) process for dynamic ownership management. Their focus was on, both cross-user file-level and inside-user block-level data deduplication. The PoW scheme guarantees tag consistency and mutual ownership verification to achieve file-level deduplication, with an asynchronous update strategy for efficient ownership management. A user-aided key approach to minimize key storage space was used for inside-user block-level deduplication. Their security and performance analysis ensured data confidentiality, tag consistency, and efficient data ownership management.

Proposed System and Methodology

As per the literature survey done and the comparative analysis, in section III, we define our problem statement to propose the system and its methodology.

Problem Definition

The problem is to detect the duplicate copies of previously uploaded data and further remove it to save the storage space on the cloud. We proposed a de-duplication scheme at client side which ensures that no duplicate data is uploaded to the cloud storage provider using convergent key encryption. We will also use technique that will consider content inside the file for de-duplication check. This technique will provide better accuracy for detecting duplicate data.

Proposed System and Methodology

The proposed system should be design in such manner that, it will eliminate limitations of traditional system. So, the objective of our proposed system are as follows:

- To develop a system which eliminates the duplicate copies of the data uploaded to the cloud storage.
- To improve the de-duplication ratio as compared to the existing system.
- To provide dynamic ownership to the user.

The proposed generalized methodology shown in figure 3, the first step is data classification further propagated for data reduction. Compression and deduplication are two different techniques for reducing the storage space required for data. Compression works by encoding data in a more efficient way, while deduplication eliminates duplicate data by storing only a

single instance of each unique piece of data. The application of both techniques depends on the type of data being stored and the intended use case. Compression is better suited for data that is not easily deduplicated, such as media files, executable code, or encrypted data. Compression can often achieve significant reductions in file size without sacrificing data fidelity. However, compression can be computationally expensive and may result in slower data access times. Deduplication is more effective for data that has a lot of redundancy, such as backups, archives, or virtual machine images. By storing only unique data, deduplication can greatly reduce the amount of storage space required. Deduplication can be faster and more efficient than compression, but it requires specialized software and hardware to be effective.

To decide whether to use compression or deduplication, is a question left to individual discretion, as compression is applied on file level and deduplication can be applied both on file level and block level. We use both compression and deduplication methods for specifically provide the benefits of both the techniques, with the scope limited to textual and image data. As shown in the proposed methodology for data reduction, we propose to perform both compression or deduplication, whereas researchers have concentrated either on compression or deduplication.

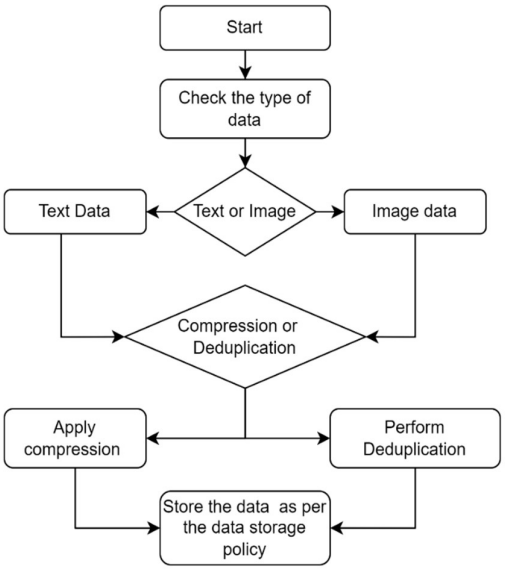

Figure 3. Proposed Methodology for data reduction.

EDAA System

EDAA (Elimination, Dynamic Ownership, Authentication, Authorization) is responsible to perform following functionalities:

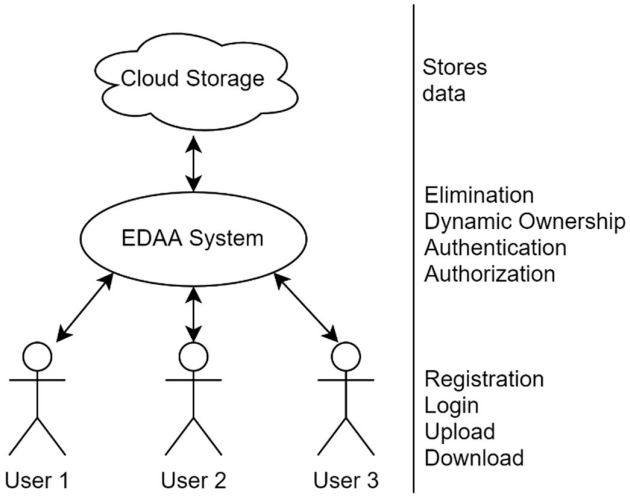

Fig 4: EDAA Architecture Diagram

- Elimination: It eliminates the duplicate files from being uploaded to the cloud instead return the pointer of the location of the original file on the cloud.
- Dynamic Ownership: It allows user sharing the same data and prove that each user sharing data owns that data in robust way. A set of data owners who shares the information on cloud as an ownership group. Each member of the group shares same group key.
- Authentication: It is responsible to give access to the trusted and verified users to access the system.
- Authorization: It allows the authenticated user to access the services of system.

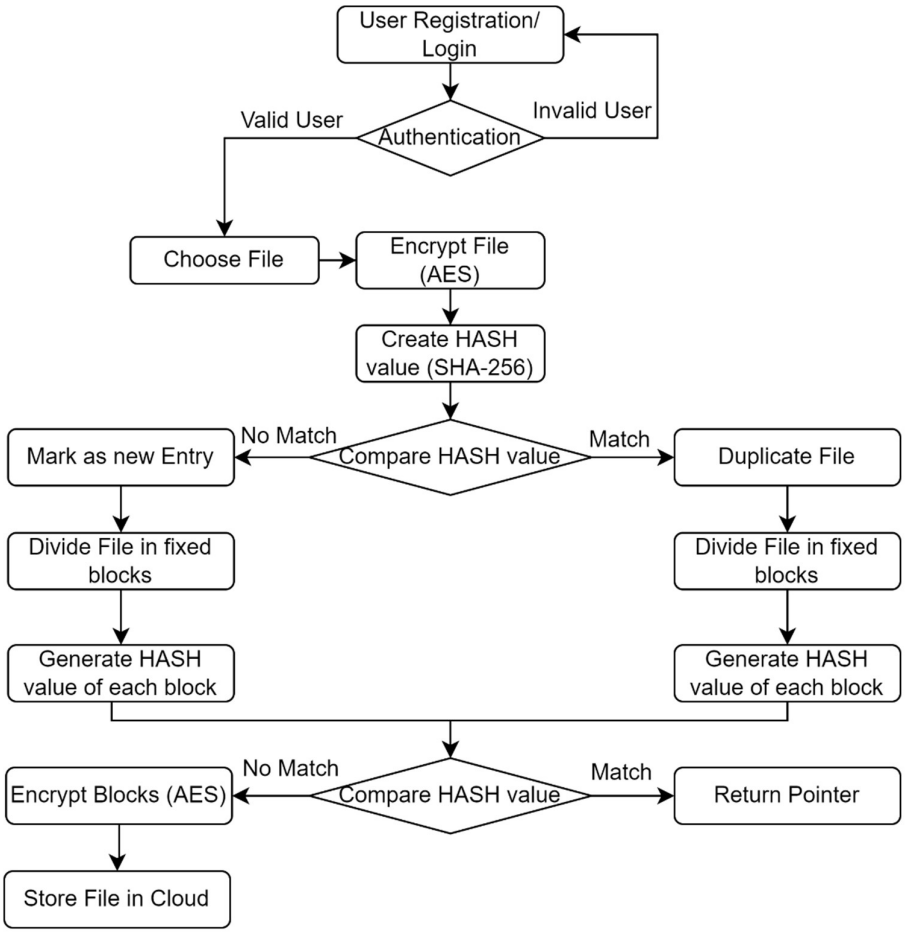

Fig 5: System Flow Diagram

Figure 5 shows the detailed insight into the system and its overall flow. The techniques used are encryption using AES-256 algorithm in conjunction with SHA-256 to create the HASH values so that they can compared for verify presence of existing blocks to lookout the duplication or redundancy of the blocks. The first comparison of HASH values is for file level duplication. Even if the file is already existing, we do second check to find the block level duplication. Encryption is done after every HASH value calculation.

Downloading and Uploading files

In this module, users can register or login to the system and upload the files and retrieve when required.

Methodology for uploading file:

1. User registration and login.
2. Request the system to check de-duplication.
3. Upload the file on the cloud.

4. Select the roles to share the uploaded file.

Methodology for downloading file:

1. Login to the user account.
2. Select the appropriate file to download.
3. Download the selected file.

Algorithms used:

Data Reduction using compression and deduplication

Given below is the algorithm and the flowchart for the workflow of the system.

Algorithm:

Step 1: Start

Step 2: Check if the data is a plain text or an image.

Step 3: Check if compression or deduplication can be applied.

Step 4: Apply compression or perform deduplication.

Step 5: Store the data thus compressed or deduplicated at appropriate storage location governed by the data storage policy after data classification in phase I and after wards data storage placement location as prescribed in phase II.

Methodology for duplication check:

The methodology based on file level and block level has three use cases with respect to File name and File content.

CASE1: Files with different name and different content.

Steps of storing file:

1.1. File is divided into fixed number of chunks. Size of chunks depends on the size of file.

1.2. Hash value of each chunk is generated using SHA-256 algorithm.

1.3. Hash value is stored in database.

1.4. Blocks are outsourced to the cloud.

CASE2: Files with same name but different content.

2.1 In this case user is prompted to change name of file. And

2.2 Go-to Case 1 step 1.1

CASE3: Files with different name but same content

3.1 If hash value of file is same,

3.2 Prompt user that the file already exists and return pointer of previous uploaded file is returned to user.

Methodology for Encryption:

a. Secure Hash Algorithm (SHA) is hashing algorithm which is widely used for information security. SHA-224 and SHA-256 are two types of algorithms in the SHS standards. We use SHA-256 in EDAA system for generating hash values of blocks used for deduplication.

b. Advanced Encryption algorithm (AES) is a Symmetric-key block cipher algorithm is used in EDAA system for encryption of files.

Cloud Storage

Cloud storage is a model of data storage in which the digital data is stored in logical pools, the physical storage spans multiple servers, and the physical environment is typically owned and managed by administrator

Results

To test the EDAA System, we took some random datasets of different sizes. We tried to upload same datasets twice and we found the results as compared to compressed data which is shown in the table below:

Table 2. Datasets

Original Size (in KB)	Compressed Size (in KB)	De-Duplicated Size (in KB)
28255	27554	14117
148982	148288	74481
8205	7512	4101
108535	107836	54257
48494	47788	24234
128554	127850	64257
88054	87356	44013

From the above table we can say that EDAA System saves almost 50% of space consumed by

the original data and nearly 45% of space as compared to the compressed data.

The above table can be represented in the graphical format which shows how effective the system saves the space consumed as compared to data compression.

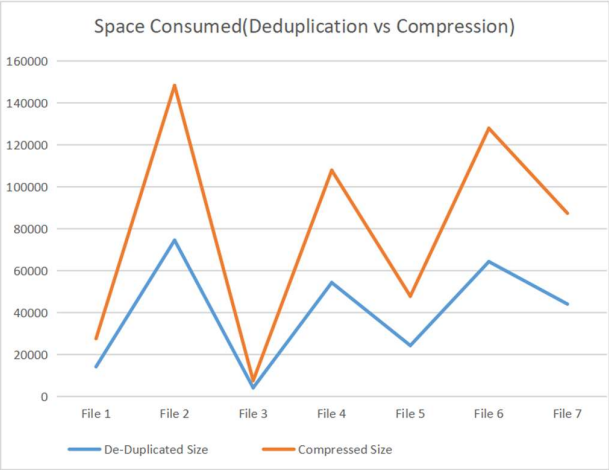

Fig 6: De-duplication vs Compression

4. CONCLUSIONS

Effective data reduction techniques play a vital role in addressing the challenges associated with the massive volume of data generated by IoT devices. These techniques offer a range of benefits, including optimized storage utilization, and reduced communication bandwidth, The selection and customization of data reduction techniques should be based on the specific requirements of IoT applications. Data reduction techniques, specifically deduplication and compression, provide effective strategies to address the challenges posed by the ever-increasing volumes of data. By eliminating redundancy and reducing data size, these techniques offer benefits such as storage space savings, improved data transfer efficiency, and enhanced system performance. We have proposed and implemented EDAA (Elimination, Dynamic Ownership, Authentication, Authorization) system to securely upload and download the data and simultaneously applying the data reduction techniques which includes, compression and deduplication. We have examined the file level, block level and content level techniques for the de-duplication of the data. In data de-duplication process, we divided that data into equal number of chunks, checked for has values and if found duplicate, skipped those chunks to be stored to avoid duplication. The system also supports dynamic authorship for the users. Our experimental results show that deduplication is more effective when it comes to storage reduction. When used in combination then the results are more optimized.

5. REFERENCES

[1] David Reinsel, John Gantz, and John Rydning. “Data age 2025: The evolution of data to life critical. Don’t focus on big data; focus on the data that’s big”. International Data Corporation (IDC) White Paper (2017)

[2] P. Agrawal, D. Puthal, N. Kumar, et al., "Data Reduction Techniques for Internet of Things: A Comprehensive Survey," J. Network Comput. Appl., vol. 103, pp. 90-106, 2018.

[3] M. Kumar, R. Nath, "Data Reduction Techniques for Internet of Things: A Review," J. Network Comput. Appl., vol. 130, pp. 19-31, 2019.

[4] D. D. Patil, N. A. Londhe, "Data Reduction Techniques for IoT Sensor Data: A Review," Int. J. Comput. Sci. Inf. Security, vol. 15, no. 7, pp. 199-203, 2017.

[5] N. Sharma, J. Singh, S. K. Aggarwal, "Data Reduction Techniques for Wireless Sensor Networks in Internet of Things: A Survey," Int. J. Comput. Appl., vol. 160, no. 2, pp. 25-30, 2017.

[6] M. S. Islam, Q. H. Mahmoud, "Data Reduction Techniques for Efficient IoT Data Management: A Survey," Future Internet, vol. 10, no. 3, Article ID 23, 2018.

[7] Deepu S R, Bhaskar R and Shylaja B S "Performance Comparison of De-duplication Techniques for Storage Cloud Computing Environment", 2014.

[8] A. Venish ,K. Siva Sankar "Study of Chunking Algorithm in Data De-duplication", 2016.

[9] Prerana T. Nitnaware, Vikas B. Maral, "Secured Authorized Block Level Deduplication on Hybrid Cloud", IJCSN International Journal of Computer Science and Network,2016.

[10] Giridhar Appaji Nag Yasa, P. C. Nagesh "Space Saving and Design considerations in Variable length De-duplication", 2012.

[11] Namrata P. Kawtikwar, M.R.Joshi, "Data Deduplication in Cloud Environment using File-Level and Block-Level Techniques", IJIR 2017

[12] Akhila K, Amal Ganesh Sunitha C, "A Study on Deduplication Techniques over Encrypted Data", 2016.

[13] K Archana, Mrs. Shubangini Patil, "Secure Data Deduplication with Dynamic Ownership Management in Cloud Storage", IJTSRD May-Jun 2018.

[14] Z. Xue, H. Qian, L. Shen and X. Wu, "A Comprehensive Study of Present Data Deduplication," 2021 IEEE 23rd Int Conf on High Performance Computing & Communications; 7th Int Conf on Data Science & Systems; 19th Int Conf on Smart City; 7th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys), Haikou, Hainan, China, 2021, pp. 1748-1754, doi: 10.1109/HPCC-DSS-SmartCity-DependSys53884.2021.00257.

[15] G.Sujatha1 , Dr.Jeberson Retna Raj2, "A Comprehensive Study of Different Types of Deduplication Technique in Various Dimensions", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 13, No. 3, 2022

[16] Wen Xia, Hong Jiang, Dan Feng, Lei Tian, “DARE: A Deduplication-Aware Resemblance Detection and Elimination Scheme for Data Reduction with Low Overheads”, IEEE TRANSACTIONS ON COMPUTERS, VOL. 65, NO. 6, JUNE 2016

[17] S. Jiang, T. Jiang and L. Wang, "Secure and Efficient Cloud Data Deduplication with Ownership Management," in IEEE Transactions on Services Computing, vol. 13, no. 6, pp. 1152-1165, 1 Nov.-Dec. 2020, doi: 10.1109/TSC.2017.2771280