Ph.D. Research Proposal

Doctoral Program in "Department Name"

LDSO-SDA: Lightweight Deduplication and Storage Optimization in Secure Data Analysis for HDFS Environment

by

<Name of the Candidate>

<Reg. No of the Candidate>

<Supervisor Name>

<Date of Submission (DD MM 20YY>

I. INTRODUCTION / BACKGROUND

Deduplication is a specialized compression technique where we can removes copies of duplicate data. This technique improves storage optimization. Data deduplication needs to be performed before inserting data into a database. The merits of deduplication are (1). It reduces storage space, (2). Improves network bandwidth and (3) Reduce overall storage cost. At the same time the limitations of deduplication technique are impact on storage performance, loss of data integrity, backup appliance issues and privacy and security [1]. Generally, there are four major steps invoked in data deduplication. When using chunking, original file splits the incoming large data into small files which is called "Chunks". Then the unique hash value is computed and assigned to each chunk for avoid hash collisions. If the new incoming chunks are existed in the database, the redundant chunk will be deleted; else the new chunk will be stored with the unique identifier [2]. The basic workflow of data deduplication is follows: chunking (fixed/variable size), hashing (SHA-1/SHA-256/SHA-512), indexing, further compression (LZ/Delta), and finally storage management (delta compression, delta restore, garbage collection, security, reliability) [3]. Hash based data deduplication method is widely used nowadays. A hash is a bit string that refers the file processed (e.g. 128 bit for MD5 and 160 bit for SHA-1) [4].

Data deduplication also suffers from another access control problem. It isn't managing encrypted data, especially for data protection from attackers. However, the encryption based deduplication does not improve the deduplication ratio. When the data is encrypted, query performance is decreased during data retrieval [5]. Data routing algorithms are proposed to assign data chunks to storage nodes. This will improve the storage space against the traditional deduplication schemes. When the name node leads to failure due to the overhead, it will fail to assign the data to data node for storage [6]. File similarity identification is another way to perform deduplication where we can find how much duplicated data exist in target files. For that in [7] authors have proposed partial hash information string algorithm (PHISA). However, finding the similarity between data files can cause a high overhead in terms of the capacity of disk storage and processing time. The primary issues involving during the deduplication are

addressed in [8] that are data granularity and storage type. Data granularity is how large are the files or the chunks of files required to be deduplicated and what's size of the chunks whether to equal size blocks or variable size blocks whereas storage type is also a major consideration in deduplication. The primary storage systems are local hard drive to memory card/SSD in the active networking devices. Adaptive block skipping is the technique that permits the incoming data chunks which could be skipped and regarded as non-duplicates directly instead of actual expensive duplication detection. This technique is insufficient to find the complete duplicates in the system [9].

1.1 Research Outline & Scope

1.2 Research Objectives

1.3 Applications / Use cases

II. RESEARCH GAPS

2.1 Common Problem Statement

2.2 Problem Definition

Naresh et al. [2] proposed a differential evolution (DE) approach based Two Thresholds Two Divisors (TTTD-P) CDC algorithm which reduces the number of computing operations. A single dynamic optimal parameter divisor D with optimal threshold value exploited the multi-operations nature of TTTD. In order to find more duplicity in the data, authors have proposed an enhanced version of TTTD and TTTD-S algorithms. It highly increases the duplicate content by using the optimal parameter. Thus the differential evolution based bucket indexed optimized data deduplication obtains a chunking speed of 16 times faster than Rabin based CDC, and 5 times faster than AE CDC and 1.6 times faster than FAST CDC with reduces 93.75% of the index lookup disk I/Os for distributed storage systems

Problems

- The security challenge of hash based deduplication isn't avoided and indexing the metadata problem is still remains.

Proposed Solution

- The security challenge of the proposed system is solved using checksum based hash value computation

Authors et al. [3] proposed a near-exact and scalable deduplication system called SiLo that sufficiently exploits similarity and locality of data streams to obtain high duplicate elimination, low RAM overhead and throughput. The major notion of this SiLo is to attains high similarity by grouping strongly correlated small files into a segment and segmenting large files and to leverage the locality in the data stream by clustering significantly continuous segments into blocks to acquire similar and duplicate data missed based on the probabilistic similarity detection.

Problems

- Deduplication ratio efficiency is poor due to the varied chunk size. The chunk size is closely related to the deduplication efficiency. If the chunk size is too big, many duplicate data will not be detected. However, the chunk size cannot be too small since small chunk size means more chunk fingerprint calculation and index searching and this will definitely lower the processing speed

Proposed Solutions

- Deduplication ratio is improved due to the fixed chunk size

Yongtao et al. [4] presented a low latency in-line data deduplication file system (LDFS). It decouples the unique data block and fingerprint indexing by using the address of data blocks to the corresponding fingerprint index and file recipe by path of read operation. To guarantee the system write performance LDFS employs finer granularity lock to optimize the block flushing strategy of write buffer. It can directly read data block from disk without accessing fingerprint index in this way and it assigns global unique identifier to every ID and gets Paper counting only requires one disk read. This system differs from traditional deduplication systems which support read any bit any file and outperforms in read operations than write operations.

Problems

- To access the file (Meta data of directories and files), this paper uses tree structure. This paper does not produce much attention in the fingerprint lookup problem.
- More unauthorized users can enter into the system which depletes the storage issues over HDFS environment.
- For index lookup issue, RAM and CPU usage is very high and it does not obtained better performance.

Authors proposed deduplication optimization method [5] that reduces duplicated data by using NewSQL database system. It is a new type of database which is widely used recently and its need to improve data reliability by periodically backing up-in memory data. In order to solve this issue, authors have proposed a deduplication optimization method called DOMe for newSQL system backup. In addition, the H-store is used a typical NewSQL database system to develop DOMe method. Finally the proposed DOMe method can reduce the duplicated NewSQL backup data, improves deduplication performance and improved the deduplication throughput by 1.5 times by the pure in-memory index optimization method

Problems

- Fingerprint index consists of a chunk information whether the chunk is duplicated, the appropriate information must be written to the metadata file for database recovery but it may cause misjudgement of some duplicated data which leads to small waste of storage space

Proposed Solutions

- Our MongoDB index consists of the hash value of the chunk and it may not cause the misjudgement problem

Ramya et al. [6] proposed a SecDedoop framework which stands for secure deduplication for Hadoop environment. At first, data users are authenticated to trusted authority by using Elliptic Curve Cryptography (ECC). In this algorithm, public and private keys are generated for requesting data to data node. Then, user's original file is divided into number of chunks and each chunk textual data is splits into number of words. For each word, TF – IDF is computed where

term frequency and inverse document frequency is computed. Then optimum node is chosen by particle swarm optimization and Mapreduce.

Problems

- Data users are authenticated in trusted authority by Elliptic Curve Cryptography, which consumes more authentication time
- An optimum node is selected by PSO and MapReduce, which cannot find the optimum one since searching is not efficient in global
- It is only suited for word based input file and it does not suited multimedia contents such as audio, image and video

Proposed Solutions

- This research work proposes lightweight security algorithm in ECC family such as Edwards Curve
- The best data node will be selected by Killer Whale Optimization algorithm, which finds by best capacity

III. RESEARCH CONTRIBUTIONS

To mitigate all such issues during data deduplication, we proposed novel algorithms with innovative solutions over the Hadoop environment.

In the real world, the amount of data is increasing gradually. Various small and medium organizations will suffer from insufficient space problem. Therefore, in this work, to face this challenge for storage systems, we propose a secure deduplication system to enhance the utility of storage space in HDFS (Hadoop Distributed File System), which is currently one of the best solutions for big data analytics. Firstly, the system is the first that achieves the security for data confidentiality in Third Party Vendor (TPV) using Edwards Curve. EC algorithm generates two keys (private key and public key) for data storage and retrieval. Secondly, we put forth methodology for data deduplication. If the user wants to upload the data into the storage system, then the storage system stores the deduplicated data.

Finally, we compute the hash value for decimal value using SHA-3 algorithm. The MD5, SHA-1, SHA-2 are good at data deduplication efficiency but poor at security and anti-collision. When using this type of hashing algorithms, an attacker can create two input strings with the same SHA-1. For this purpose, we proposed a SHA-3 standard algorithm.

Index table is one of the most important components of data deduplication system. Index table is constructed using MongoDB database. Before that, we cluster the similar hash value in one index using Neutrosophic C Means algorithm and put it into MongoDB towards this we compute the "n" number of indexes.

Index is maintained in the data node by Merkle Hash Tree (MHT) algorithm. In addition, necessary algorithms have proposed for new file insertion, querying the data on HDFS. Here data nodes (DNs) are refers as slave nodes whereas the namenode (NNs) acting as a master implements algorithm to decide where to place the data. In HDFS, two types of name nodes are used: primary name node (PNNs) and Secondary Name Node (SNNs). PNN consists of deduplicate manager and storage optimizer. SNNS saved Meta data and it can rebuild a failed name node. Data received by the DO is tracked the Name node.

Utilizing this KWA algorithm, we can reduce the data access time and retrieval is made easier for the data enquired. The killer whale algorithm decides whether a block of data is new to the system. It is the first attempt when KWA based Map Reduce algorithm is applied in the field of optimum data node selection, which further improves the storage efficiency. In KWA, the fitness value is computed using memory usage of the server, node size (available space), I/O capacity and CPU power. This process improves the searching speed (maximize) and reduce the running time. The node having the maximum fitness value is chosen to hold the data. Finally, data retrieved using user's public key generated by EC algorithm.

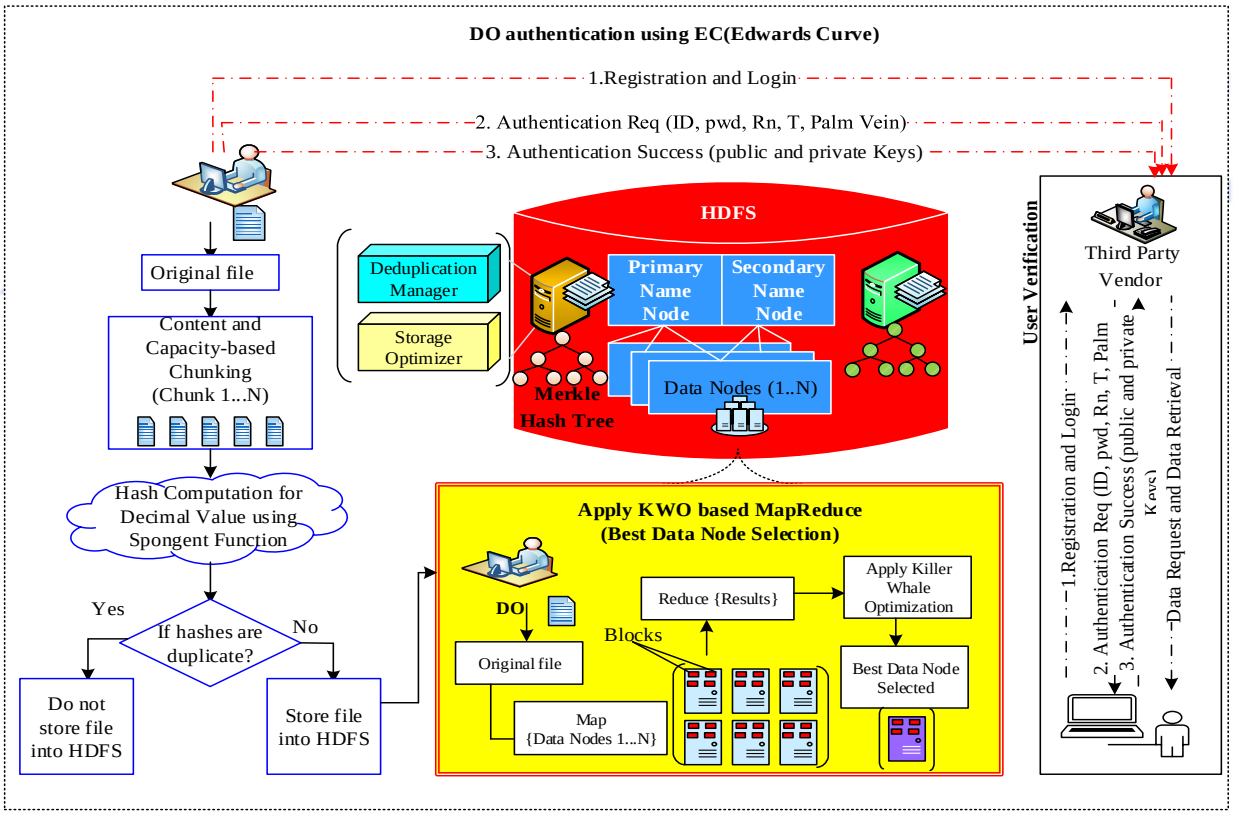
Performance Evaluation

The performance of the proposed system is evaluated in terms of the following,

- Deduplication elimination ratio (DER)
    - With respect to number of data records

- Deduplication time
    - With respect to number of data records
- Throughput
    - With respect to number of data records
- CPU power
    - With respect to number of data records
- Overall computation time
    - With respect to number of data records
- Data retrieval time after deduplication
    - With respect to number of data records

System Architecture



IV. RESEARCH NOVELTIES

- **Theorem 1.** User privacy (our proposed scheme is authenticated against malicious users). In this study, individual user privacy is often given to control the privacy degree and also get the legitimate information with other users in a massive environment.

  PROOF 1. With the use of soundness in authentication, both users and owners cannot store or retrieve the file without complete authentication. For owners, authentication requires satisfied credentials$\{ID, Pwd, B, Rn, T\}$, whereas the users cannot exceed the K-values (maximum attemptts to access data).

- **Theorem 2.** Privacy preservation (LDSO-SDA cannot leak the private data for users and owners)

  PROOF 1. With the successful authentication, we must check the data ownership verification when upload the data. When the attacker a can be submitted a false file, we check the authentication and only eligible data owners can check for deduplication.

V. PREVIOUS WORKS & LIMITATIONS

In this section, what are the works have been previously released in the field of storage optimization and secure deduplication in Hadoop environment

Paper 1

Title – Heterogeneous Data Storage Management with Deduplication in Cloud Computing

Concept

This article presented about the heterogeneous data storage management scheme, which flexibly handles both access control and deduplication management across multiple cloud service providers (CSPs). This proposed scheme supports cloud user's data privacy because the data stored at the cloud is in an encrypted form. For that identity privacy is used to apply pseudonyms in Key Generation Center (KGC) in which a real identity is linked to a pseudonym. This pseudonym is verified and certified using the KGC.

Limitations

- Data holders suffer due to the lack of privacy

Paper 2

Title – Dynamic Deduplication Decision in a Hadoop Distributed File System

Concept

This article proposed a dynamic deduplication decision to enhance the performance of storage utilization of a data center which uses HDFS (Hadoop Distributed File System) as its file system. The presented system was formulated a proper deduplication strategy to sufficiently utilize the storage space under the limited storage devices. This strategy deletes redundant and duplicate data to increase the storage space. Hence the proposed dynamic deduplication decision maker to improve the usage of storage. This system is also suitable for small enterprises and organizations, especially used in education environments.

Limitations

- Ensure data reliability without the loss of storage space and it does not improve the capacity of system

Paper 3

Title – Attribute based Storage Supporting Secure Deduplication of Encrypted Data in Cloud

Concept

This paper presented an attribute based encryption (ABE) with secure deduplication in a hybrid cloud settings. Here a private cloud is used for duplicate detection and a public cloud manages the storage. When compared with the traditional data deduplication systems, this system have obtains two advantages. First, verify the confidentiality of data based on the access policies than sharing the decryption keys. Next step is to obtain the standard notion of semantic security for data confidentiality. Generally, the standard ABE systems do not support secure deduplication, which makes the system costly to be applied in some commercial storage services.

Limitations

- When using attribute based encryption (ABE) scheme is that data owner needs to use every authorized user's public key to encrypt data. The application of this scheme is restricted in the real environment because it uses the access of monotonic attributes to control user's access in the system

Paper 4

Title – A Novel Approach to Data Deduplication over the Engineering-Oriented Cloud Systems

Concept

This article presented a duplicate less storage system on cloud computing platforms. The proposed deduplication storage system can manage find duplicate data over the cloud system. It uses two essential components such as a front-end deduplication application and a mass storage system as a back end. In this paper, HDFS refers to build up a mass storage system and used HBASE (Hadoop Database) for fast indexing. This paper developed a deduplication application, parallel and scalable deduplicated cloud storage system which is efficient and accurate for distributed and cooperative data intensive engineering applications.

Limitations

- HBase database can leads to a single node failure. If it fails, the entire system will stop working. Further, it renders any HDFS file system at that node in accessible

Paper 5

Title – Bucket-size Balancing Locality Sensitive Hashing using the Map-Reduce Paradigm

Concept

This paper proposed a locality sensitive hashing method to increase the recall rate without entailing the significance cost. However, the increase in the size of a candidate set improves the recall of similar data but decreases the processing speed. Thus the proposed method uses a

random hyperplane partitioning technique to make the buckets to which data objects are distributed. The nearest neighbors located on the other side of hyperplanes which can be false negatives if the bucket to which query belongs is observed for determining similar neighbors. Further, the over-sized buckets are split by adding additional hyperplanes to control the bucket sizes. To enhance the performance in terms of processing speed, each jobs on MapReduce framework.

Limitations

- The bit strings for the buckets can be different each other in the proposed system. When select variable size of blocks, it consumes more CPU and memory to determine data blocks boundry.
- Similarly, the locality sensitive hashing is fast and space efficient method but it slightly provides worse error bounds.

Paper 6

Title- A Novel Approach of Data Deduplication for Distributed Storage

Concept

This paper a new approach for data duplication which reduces data redundancy saves storage space and simplifies the management of data chunks. This paper makes three steps: chunking, fingerprinting and fingerprints indexing. Firstly in chunking, data files are splits into chunks and the chunk boundry is determined by the value of the divisor. Each chunk is uniquely identified using hash signature algorithms including SHA-1, MD-5 and SHA-256. To find the best value of the divisor, Genetic Algorithm (GA) is used. Secondly in indexing, each chunk's hash value is stored which reduce the searching time. For indexing, binary search tree (BST) is proposed which has the time complexity of O (long n).

Limitations

- The main focus of this paper is to obtain the high deduplication ratio. So that the proposed approach uses GA to find the best value of divisor, but it takes more time to complete the process

Paper 7

Title – LDAP: A Lightweight Deduplication and Auditing Protocol for Secure Storage in Cloud Environment

Concept

This paper presented a lightweight auditing method for data deduplication and integrity verification. This method integrates two approaches such as hashing and symmetric encryption with enhances the performance of the distributed hash table data structure. Hence, the proposed method reduces the communication and computation overhead for integrity validation and also enables efficient dynamic operations of the data. The cuckoo filters find whether a block of data is new to the system

Limitations

- The proposed method reduces the computation cost at user side, but the system still utilizes high memory and I/O capacity

Paper 8

Title – Data Deduplication Techniques for Efficient Cloud Storage Management: A Systematic Review

Concept

This paper presents a broad methodical literature review of existing data deduplication techniques along with various existing taxonomies of deduplication techniques that have been based on cloud data storage. Furthermore, the paper investigates deduplication techniques based on text and multimedia data along with their corresponding taxonomies as these techniques have different challenges for duplicate data detection. This research work is useful to identify

deduplication techniques based on text, image and video data. It also discusses existing challenges and significant research directions in deduplication for future researchers

Paper 9

Title – Study of Chunking Algorithm in Data Deduplication

Concept

In cloud data storage, the deduplication technology plays a major role in the virtual machine framework, data sharing network, and structured and unstructured data handling by social media and, also, disaster recovery. In the deduplication technology, data are broken down into multiple pieces called “chunks” and every chunk is identified with a unique hash identifier. These identifiers are used to compare the chunks with previously stored chunks and verified for duplication. Since the chunking algorithm is the first step involved in getting efficient data deduplication ratio and throughput, it is very important in the deduplication scenario

Paper 10

Title – A Comprehensive Study of the Past, Present, and Future of Data Deduplication

Concept

In this paper, we first review the background and key features of data deduplication, then summarize and classify the state-of-the-art research in data deduplication according to the key workflow of the data deduplication process. The summary and taxonomy of the state of the art on deduplication help identify and understand the most important design considerations for data deduplication systems. In addition, we discuss the main applications and industry trend of data deduplication, and provide a list of the publicly available sources for deduplication research and studies. Finally, we outline the open problems and future research directions facing deduplication-based storage systems.

Paper 11

Title – A Novel and Efficient De-duplication System for HDFS

Concept

The objective of the research is to eliminate file duplication by implementing De-duplication strategy. A novel and efficient De-duplication system using HDFS approach is introduced in this research work. To implement De-duplication strategy, hash values are computed for files using MD5 and SHA1 algorithms . The generated hash value for a file is checked with the existing file to identify the presence of duplication. If duplication exists, the system will not allow the user to upload the duplicate copy to the HDFS. Hence memory utilization is handled efficiently in HDFS.

Limitations

- Less efficiency in data deduplication system

Paper 12

Title – Encrypted Data Management with Deduplication in Cloud Computing

Concept

To preserve cloud data confidentiality and user privacy, cloud data are often stored in an encrypted form. However, duplicated data that are encrypted under different encryption schemes could be stored in the cloud, which greatly decreases the utilization rate of storage resources, especially for big data. Several data deduplication schemes have recently been proposed. However, most of them suffer from security weakness and lack of flexibility to support secure data access control. Therefore, few can be deployed in practice. This article proposes a scheme based on attribute-based encryption (ABE) to deduplicate encrypted data stored in the cloud while also supporting secure data access control. The authors evaluate the scheme's performance based on analysis and implementation.

Limitations

- ABE gives high complexity

Paper 13

Title – DCDedupe: Selective Deduplication and Delta Compression with Effective Routing for Distributed Storage

Concept

In DCDedupe, authors proposed a pre-processing step to identify content similarity and data chunks are classified into different categories. Then, the appropriate routing algorithm ensures the data chunks are sent to the right target storage nodes in the distributed system to boost the storage efficiency. Our evaluation shows that generally storage space saving by DCDedupe outweighs the performance penalties. In some use cases, DCDeupe may become meaningful to trade off some throughput with optimized storage costs. The overheads to Input/Output (IO) operation and memory usage have also been studied with design recommendations.

Paper 14

Title – File Similarity Evaluation Scheme for Multimedia Data using Hash Information

Concept

This paper proposes a novel file similarity evaluation algorithm called PHISA (Partial Hash Information String Algorithm). To evaluate the performance of the proposed system, we compare PHISA to well-known file similarity tools. The evaluation result shows that PHISA reduces the processing time and increases the similarity evaluation accuracy.

Limitations

- PHISA cause more overhead when single hadoop cluster created

Paper 15

Title – Design and Implementation of Various File Deduplication Schemes on Storage Devices

Concept

In the paper, authors aim at designing and implementing several file deduplication schemes built in the private cloud storage appliance, based on different duplication checking rules, including file name, file size, and file partial/full content hash value. Experiment results show using partial content hashing based file deduplication scheme achieves a reasonably balanced performance without overutilized limited local computational resources.

Zheng Yan., Lifang Zhang, Wenxiu Ding, Qinghua Zheng, (2017). Heterogeneous Data Storage Management with Deduplication in Cloud Computing, IEEE Transactions on Big Data, Vol. PP, Issue 99, PP. 1-14

Ruay-Shiung Chang., Chih-Shan Liao., Kuo-Zheng Fan., Chia-Ming Wu., (2014). Dynamic Deduplication Decision in a Hadoop Distributed File System, Hindawi Publishing Corporation, Vol. 2014, PP. 1-14

Hui Cui., Robert H. Deng., Yingjiu Li., Guowei Wu., (2017). Attribute based Storage Supporting Secure Deduplication of Encrypted Data in Cloud, IEEE Transactions on Big Data, Vol. PP, Issue 99, PP. 1-13

Zhe Sun., Jun Shen., Jianming Young., (2013). A Novel Approach to Data Deduplication over the Engineering-Oriented Cloud Systems, University of Wollongong Research Online, Vol. 20, Issue 1, PP. 45-57.

Kyung Mi Lee, Yoo-Su Jeong., Sang Ho Lee, Keon Myung Lee,., (2018). Bucket-Size Balancing Locality Sensitive hashing using the Map Reduce Paradigm, Cluster Computing, PP. 1-13

Shubhanshi Singhai., Akanksha Kaushik., Pooja Sharma., (2018). A Novel Approach of data Deduplication System for Distributed Storage, International Journal of Engineering and Technology, Vol. 7, Issue 2, PP. 46-52

Esther Daniel., N.A. Vasanthi., (2017). LDAP: A Lightweight deduplication and Auditing Protocol for Secure Data Storage in Cloud Environment, Cluster Computing, PP. 1-12

Raveet Kaur., Inderveer Chana., Jhilik Bhattacharya., (2017). Data Deduplication Techniques for Efficient Cloud Storage Management: A Systematic Review, The Journal of Super Computing, PP. 1-51

A. Venish., K. Siva Sankar., (2016). Study of Chunking Algorithm in Data Deduplication, From book, Proceedings of the International Conference on Soft Computing Systems: ICSCS 2015, Vol.2, PP. 13-20

Wen Xia., Hong Jiang., Dan Feng., Fred Douglis., Philip Shilane., Yu Hua., Min Fu., Yucheng Zhnag., Yukun Zhou., (2016). A Comprehensive Study of the Past, Present and Future of Data Deduplication, Proceedings of the IEEE, Vol. 104, Issue 9, PP. 1681-1710

C. Ranjitha., P. Sudhakar., K. S. Seetharaman., (2016). A Novel and Efficient De-duplication System for HDFS, Procedia Computer Science, Vol. 92, PP. 498-505

Zheng Yan., Mingjun Wang., Yuxiang Li., (2016). Encrypted Data Management with Deduplication in Cloud Computing, IEEE Cloud Computing, PP. 28-35

Binqi Zhang., Chen Wang., Bing Bing Zhou., Dong Yuan., Albert Y. Zomaya., (2018). DCDedupe: Selective Deduplication and Delta Compression with Effective Routing for Distributed Storage, Journal of Grid Computing, PP. 1-15

Byung-Kwan Kim., Su-Jin Oh., Sung-Bong Jang., Young-Woong Ko., (2017). File Similarity Evaluation Scheme for Multimedia Data using Hash Information, Multimedia Tools and Applications, Vol. 76, Issue 19, PP. 19649-19663

Kuan-Wu Su., Jenq-Shiou Leu., Min-Chieh Yu., Yong-Ting Wu., Eau-Chung Lee., Tian Song., (2017). Design and Implementation of Various File Deduplication Schemes on Storage Devices, Mobile Networks and Applications, Vol. 22, Issue 1, PP. 40-50, 2017

Naresh Kumar., Shobha Antwal., S.C. Jain., (2018). Differential Evolution based Bucket Indexed Data Deduplication for Big data Storage, Journal of Intelligent & Fuzzy Systems, Vol. 34, PP. 491-505

Wen Xia., Hong Jiang., Dan Feng, Yu-Hua., (2015). Similarity and Locality based Indexing for High Performance Data Deduplication, IEEE Transactions on Computers, Vol. 64, No. 4, PP. 1162-1183

Yongtao Zhou., Yujui Deng., Laurence T.Yang., Ru Yang, Lei Si, (2018). LDFS: A Low Latency In-Line Data Deduplication File System, IEEE Access, Vol. PP, Issue 99, PP. 1-10

Longxiang Wang, Zhengdong Zhu., Xingjun Zhang., Xiaoshe Dong, Yinfeng Wang, (2017). DOMe: A Deuplication Optimization Method for the NewSQL Database Backups, PLOS ONE Research Article, Vol. 12, Issue 10, PP. 1-17

Ramya, P. & Chinnasamy, Sundar. (2020). SecDedoop: Secure Deduplication with Access Control of Big Data in the HDFS/Hadoop Environment. Big Data. 8. 147-163.