ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 7, Issue 4, April 2019

An Efficient Cloud Storage through Data Deduplication Process Using Machine Learning Techniques

Mrs.S.Ruba, Dr.A.M.Kalpana

Assistant Professor, Department Computer Science and Engineering, Government College of Engineering, Salem(D.t), India

Professor/Department of Computer Science and Engineering, Government College of Engineering, Salem(D.t), India

ABSTRACT: Cloud computing is an integral part of computer science resources, impacts data processing, is primarily used as data storage, and does not require constant active management by customers. Clients upload input files to the cloud for data storage and maintenance. By growing their digital library from cloud storage, providers can be impacted by database replicas, resulting in lower bandwidth and larger storage space requirements resulting in lower performance and higher operating costs. To do. Cloud Storage Providers (CSPs) use enhanced machine learning techniques index scores to analyze replicated data. The proposed system focused on her two methods of cloud data: data and data deduplication process. We use advanced machine learning (ML) algorithms to make this process efficient and enable efficient cloud storage. A proposed semi-supervised multi-label filtering technique (SMFT) is used to split the data file, splitting it on block boundaries determined by divisor values. Each can have a unique identification value that is computed and stored via an index. A semi-supervised technique defined by training and test data split by a filtering technique with multiple labels in file format is processed. The Chunk Data Streaming Deduplication Analysis (CDSDA) ML algorithm data deduplication is essential for analyzing user-uploaded data against data stream index values for deduplication. Compare the data deduplication of the CDSDA proposed algorithm to the index value to detect redundancy in cloud data. Experimental results of ML techniques help cloud users to efficiently use cloud data storage space by avoiding storing duplicate data and saving bandwidth.

KEYWORDS: Cloud Computing, Machine Learning, Chunking process, SMFT, Deduplicated Process, CDSDA, and efficient cloud storage.

I. INTRODUCTION

Cloud computing becomes more powerful from the latest years and data storing is very important in the research field. Cloud computing arise a combination of traditional networks and computer technology. One of the main concepts of cloud computing reduces the data processing burden on the client terminal system through the sequence to enhance handling capacity. There are high storage capacity and bandwidth are the main focused feature in the cloud. The Cloud storage provider has to reduce the redundancy of data and an empty file of data by avoiding data deduplication.

The database is an essential source for each organization and that can be derivative from various sources. Each heterogeneous have a different indication for the same entity that leads to being a duplication in the repository. Due to duplication many problem are occurred as follows:

- **Loss of Quality:** The duplicated and other inconsistencies of data lead to falsification of the reports and distorted conclusions.

DOI: 10.15680/IJIRCCE.2019. 0704043

- **Performance degradation:** Due to addition unused data demand a more processing method and required more time to answer simple queries.
- **Increased cost**: Because of unused space in cloud data which means the additional size of useless data may require more storage spaces for the useless data also users should pay an additional amount.
- **Required more spaces**: By adding repeated data required more cloud spaces and also the empty size of data also fill the spaces.
- **Time Complexity:** Further computational process power and response time level is high while uploading data into the cloud.

Machine learning (ML) is used to provide learning techniques and focused on developing a computer program that can teach themselves by growing and modifying to exposed new data. The attribute term generally refers to record fields table attributes, data items, etc. This ML approach is applied to both chunking and index term matching. During the learning phase, the mapping rule and the transformation weights are defined. The detection and removal of these duplicated records from the data repository are called record deduplication. Deduplication is done by machine learning techniques which improve the integrity and quality of data and also reduce the effort and cost of obtaining data. Data deduplication of the ML algorithm is used for eliminating the duplicated data. In the deduplication process is compared with the index term.

Chunking is the best idea for partitioning a large amount of data into small data chunks, this will easily able to recognize the chunk data and modify the data efficiently. The chucking method offers several features of statistical calculation, range queries, and keyword searching. Chunking is processed by the various method of file level, content level, size level, static level, and so on. The research focused on chunking and deduplication of cloud data for efficient storage and easy query searching process.

Data deduplication is easily analyzed by the partition of data which was stored in the index term of a repository. The cache memory will save the keywords of hash value by partitioning of cloud data through various techniques. In the deduplication process, the unique value of data chucks is identified and stored in the repository of index values. And the user uploaded data are compared with the similarity matching of index terms of unique values by using advanced machine learning algorithms.

II. RELATE WORK

ShubhanshiSinghal et al., Due to the expansion growths of digital data, the data duplication has become standard components of a modern system. The paper has performed with three stages: chunking, fingerprint, and indexing fingerprints. In chunking data files are divided into chunk boundary is decided by the divisor values. For each one there may be a unique value of identification that is computed through hash signature (i.e. MD-5, SHA-1, SHA-256), which are called a fingerprint. Finally, fingerprints are stored as an index term to detect replication chunks by comparing the same fingerprints. The Genetic algorithm (GA) is applied to perform a better function of value division. Binary search tree (BST) based on the indexing time complexity. Which is least among the various searching algorithms.

Chi Yang et al., Big data sensing is established in both the scientific research and industries where the data have been present high velocity and volumes. Cloud computing offers a better platform for big data sensing for data processing and provides high storage and flexible software services. At present big data processing is adopted some compression techniques of data from cloud storage. However, lack of efficiency and scalability of traditional data processing and data compression due to the large volume and velocity of sensing data. This paper has been proposed a novel scalable data compression method based on the comparison of similarity among the classification of data chunks. MapReduce algorithm is used to implement the chuck partition and scalability of data.

Thanh Nguyen et al., The survey has been proposed for cloud-based storage difficulties of Big File Cloud (BFC) by using the advanced technique of storing key values. Many problems occur while designing an effective storage engine system with requirements such as low latency, large file processing, lightweight metadata, deduplication, scalability, parallel input and output process, distributed process. By loading heavy data in the cloud the key values are useful to perform relational databases. The survey paper applies the key-value stores with the automatic increment of integer key for solving a problem for big file storing. The results can build scalable distributed data in the cloud storage that supports a file size of terabytes. The proposed systems have a scalable compression method based on the chunk to improve efficient compression.

Anju K S et al., Large dataset of recommendation system and medical datasets are imposed on a challenging quality of semantic duplicates. Deduplication is necessary for a large number of datasets to reduce the duplication of data. To enhance storage capacity dramatically to optimize innovation techniques. To run a sensible time required sufficient ample and find many copies with a solid method to identify the duplicate copies. Bloom filter is a proposed method to detect similarity-based copies to the exploited inside of files. Bloom filter is used to analyze the portion of duplicated data where the storage space of copies savings will probabilistic to decrease the search time. To enhance the good performance of the system of the Levenshtein distance algorithm is used to find the similarity between the chunks.

GuangpingXu et al., a Learning-based data deduplication algorithm is called LIPA, which uses reinforcement learning to build an adaptive index structure. The existing system required to analyze hash function to analyze the deduplication often and often which are critical and time complexity for data deduplication. In the previous bottleneck, problems have occurred while backup large scale of data by chunk lookup disk to inline deduplication method. The entire chuck index hard in RAM and directly affected the dependent sampling ratio of deduplication. Our learning method requires only a little memory to store an index value to achieve even or the same deduplication ratio in the existing method. In our proposed method data stream is used to broken relatively. Using reinforcement learning the feature are represented to score the similarities of this segmentation. Our experimental results illustrated our method essentially reduces memory storage and spaces and effective deduplication.

III. PROPOSED METHODOLOGY

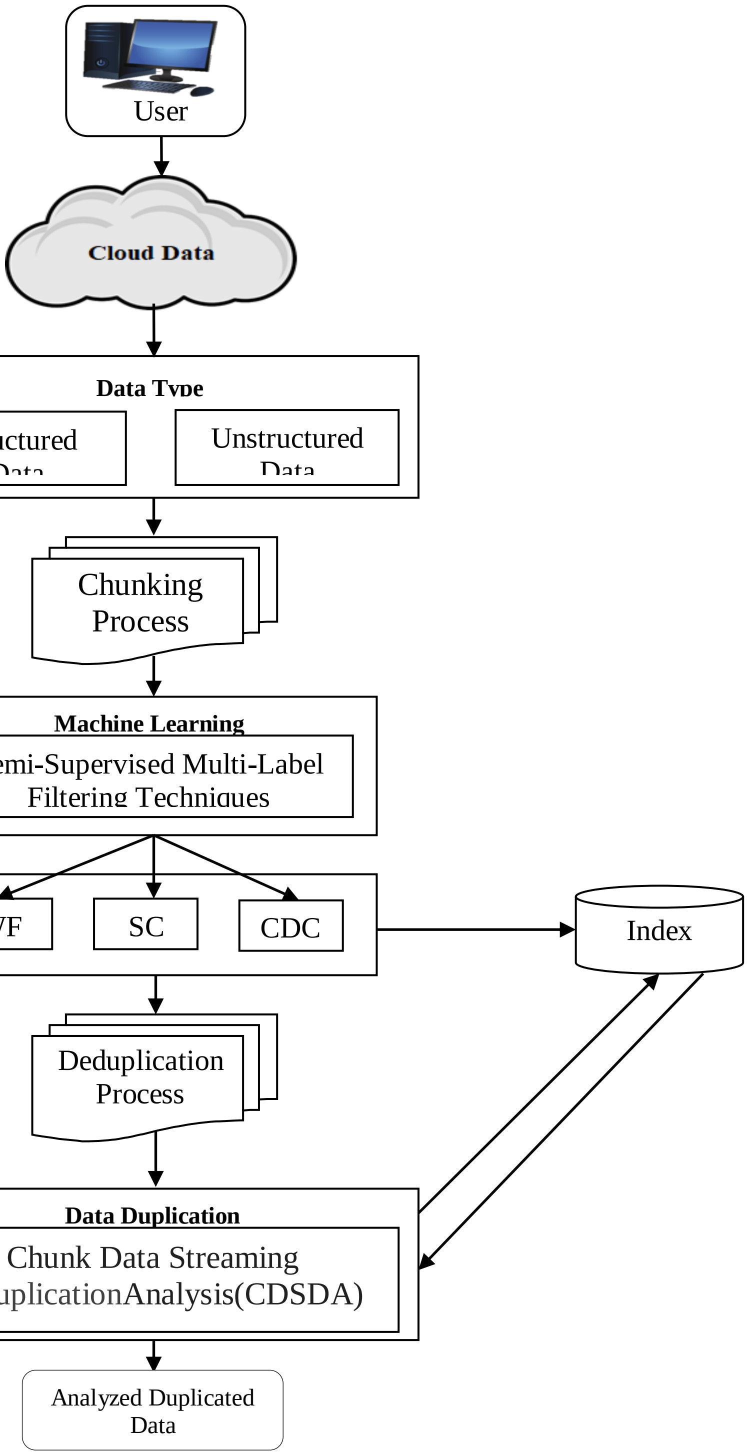


Figure 1: Overall Proposed Methodology

A) Cloud Computing

Cloud computing offers a better infrastructure for large data processing with power computation, scalability, the capability of storage, resource reuse, and essential for alignment processes. The cloud data can be classified into two types structured and unstructured data. Structured data is partially organized and that will be formatted in an easy way of searchable techniques in relational databases. Unstructured data has no sequence organization or no pre-defined format and difficulties to analyze a process, and collect the data. Due to a large amount of repeated data the computing device of cloud storage platform, upload bandwidth pending challenge and time taken for data backup. In the proposed system of ML algorithm used to improves the data deduplication by exploiting the chunk method. Further combined user and cloud to detect the duplicated data with good balancing between cloud storage and chuck mainly used deduplication time reduction.

B) Chunking Process

Data chunking system having a great impact on the effectiveness of data deduplication. Among the relationship of the deduplication ratio, the average value of the chunk size is efficient. Chunking a large amount of data for easy process and time saved by using partitioning of data. However, Each chunk was assigned by an identifier by using the index term for example for each block unique ID is generated. The comparison will take place between the particular unique ID and identifier. If the unique ID is already present in the index term then it indicated the data was deduplicated. Therefore this technique is used to save the previously stored data.

i) Semi-supervised multi-label filtering techniques (SMFT)

Chunking process is one of the learning techniques of machine learning which learn by its experience to perform the chunking technique. Using chunking mechanism results is produced a subgoal of big data files. The semi-supervised multi-label filtering technique used to partition the data depend upon the type of file level, size, content-based by using this type of various technique the large data are chuck using these SMFT techniques. They are multi-label of data types that are splitting from both structured and unstructured data. The small files are filtered first depend upon the size for efficient reasons and large data streams are broken by various chunks using this type of filtering technique.

ii) Partition of Chunking

Chunking is a process of splitting a large file into a block-wise file called chunks. Chunks are defined as new products to forms small steps of assignments that are grouped into larger steps of assignment. Data chunks are defined as the same types of file are deduplicated into a partition of chunk and stored in the hash engine and perform to analyze data replication. Efficient chunking is one of the key elements that will decide an overall process of deduplication performance. The below figure shows the partition of chunking by using the machine learning process.

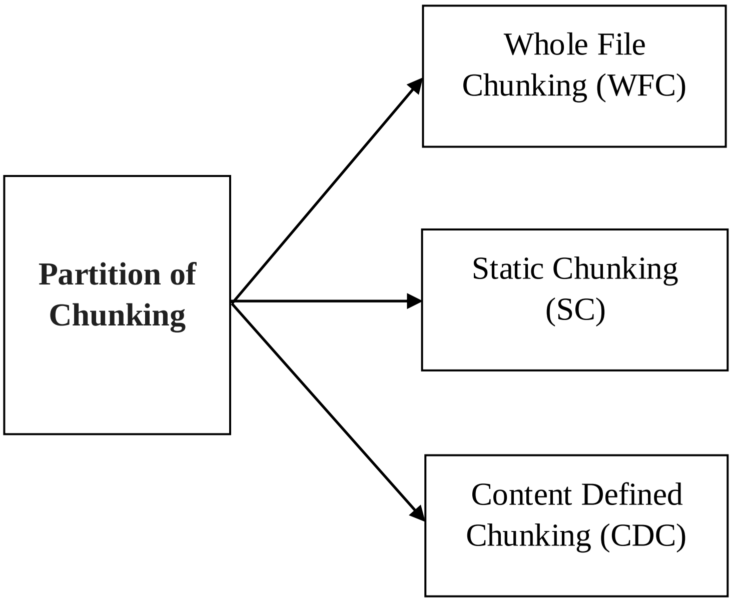


Figure 2: Partition of Chunking Techniques

a) Whole File Chunking (WFC)

It is considered a complete file as a chunk, rather than breaking larger files into multiple chunks. In this method, only one index value is created for the entire file and the similar is compared with already stored in whole file indexes. Chunking file can be divided into three categories depending on the statically uncompressed file type, dynamic uncompressed file, and compressed files. Compressed files are chunked by whole file content based on the chunking file with more extension. After data chunking, unique data will be stored in the index term of cloud storage with the parallel management. Using the machine learning algorithm of filtering techniques is used to partitioned the data and stored index terms of structure.

b) Static Chunking (SC)

Static files are fixed-size chunking by SC with an ideal chunk process. The deduplication is efficient for data chunking systems to differ among various applications. Whether SC can perform and divided the files into two major categories: static and dynamic files. The static is defined ad unmodifiable that is common and dynamic files are modifiable at a run time process. To strike a trade-off for better elimination of duplication ratio and duplicated overhead. The static chunk is fully analyzed and using the ML algorithm of SMFT and stored in cache memory as an index term.

c) Content defined Chunking (CDC):

CDC technique is one of the major algorithms for partitioning techniques it will need to check every byte of a stream into input data by searching chunk boundaries. This execution of partition time is proportional to the approximation of the number of input data streams. The chuck of CDC is used to content-based partition techniques which are used to split the content by analyzing whole data. And the hash value of the index term is stored in the repository of index term using the ML algorithm of a multi-label process.

$$S = \sum_{x=1}^{n} . \sum_{y=1}^{n} P(x,y)^2$$

Where,

S: is defined as a dataset of input data.

DOI: 10.15680/IJIRCCE.2019. 0704043

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 7, Issue 4, April 2019

x and y: the summation of training and testing set entire dataset.
n: Number of content data
P: processing the chunk values.

$$C_{ch} = \frac{[\Sigma_x \cdot \Sigma_y P(x,y)]}{\mu_x \mu_y}$$

Where C_{ch}has indicated the chunk of entire data and Patitition of data into x and y. Analyzing the whole dataset and the 'μ' defined the content-based partition of data the 'n' is a number of counts that are stored in the index term.

Proposed Algorithm (SMFT)

Input:String S, S is a streaming data, **string**, P partitioning of data Chunk

Output:Partition of Chunking

```
1) Declare Variable
2) Initialize variable
3) Read Streaming data and assign into variable S
4) If S != Null then
       goto step 5
   Else
       Check the given streaming data.
   End if
5) Check the streaming data based on Filtering Technique
   If P != Null then
       If S == P then
           goto step 4
       Else
           temp db  ←S
           classification model ← read file
           index ←store model
       End if
   Else
       index ←db
       temp db ← index value store
       P ← temp db
       goto step 5
   End if
6) End
```

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 7, Issue 4, April 2019

C) Data Deduplication Process

Data deduplication is used to eliminate the redundancy of data cloud service provider target to deduplicate due to the former's capability fore essential for reducing the amount of data transfer over the communication bandwidth. Data deduplication an effective data compression method that exploits data replication, a partition of large data into smaller blocks called chunks that represents these chunks by their WFC, SC, and CDC. The content replaces the redundancy chunks with their content after chunks the complete data with the index lookup, and store the unique chunk value for analyzing redundancy for efficient storage.

Data deduplication is needed, for the following reasons. They are,

- To reduce the utilization of storage and
- To improve efficiency in the Database.
- Minimum operational costs and high performance.

i) Machine Learning algorithm of CDSDA

This research proposed aChunk Data Streaming Deduplication Analysis (CDSDA) algorithm is carried out whole static content based deduplicate a streaming data. In this approach, the content-based deduplication of the chunk data will allow the data to be scanned intelligently at the same time and a unique index value is stored for further usage. Also, streaming data compared with the index term to analyze deduplication using a machine learning algorithm of the CDSDA algorithm.

$$Q = \sum_{x=0}^{n-1} . \sum_{y=0}^{n-1} \frac{P(x,y)}{i - (x+y)^2}$$

Where,
x and y: Cloud dataset
P(x,y): Partition of Data
i: storing data of Index term

$$D = \sum_{x} . \sum_{y} P_d\ [x,y] = \ InP_d[x,y]$$

The Partition of chunk data of x and y and the index term of user input data are compared with the index term with unique ID of chunk.

$$R = \sum_{x=1}^{n-1} . \sum_{y=1}^{n-1} P(x,y)(x-y)^2$$

Data efficiency is based on a data process which means it should process by avoiding redundancy and robustness to improve throughout data streaming. Most of the application has a huge amount of data it cost on storage and ability to reduce the storage and query timing on the database. Data duplication is an optimistic method to a conception of resources, improves storage capacity in the database.

DOI: 10.15680/IJIRCCE.2019. 0704043

Algorithm: Machine learning algorithm of CDSDA

Input: A segment of the incoming data stream, seg;

Output: A container buffer, buf.

```
input data ∗i = features of segment(seg);
/* lookup the feature in the context table */
if is in dataset(d, i) then
    /* choose proper segments from recipes into cache */
    fingerprint ∗ch = choosing champions (f);
    cache prefetch(ch);
    if evict recipe from cache() then
        feedback to context data();
    end if
end if
/* check each index term in order */
while seg =∅ do
    index ∗i = get index from chunk(ck);
    chunk ∗data = get chunk from segment(seg);
    if is in cache(i) then
        /* the chunk is duplicate and analyze deduplicate */
        update to context data(i,data);
        write duplicate chunk(i,buf);
    else
        write unique chunk(i, data, buf);
    end if
end while
```

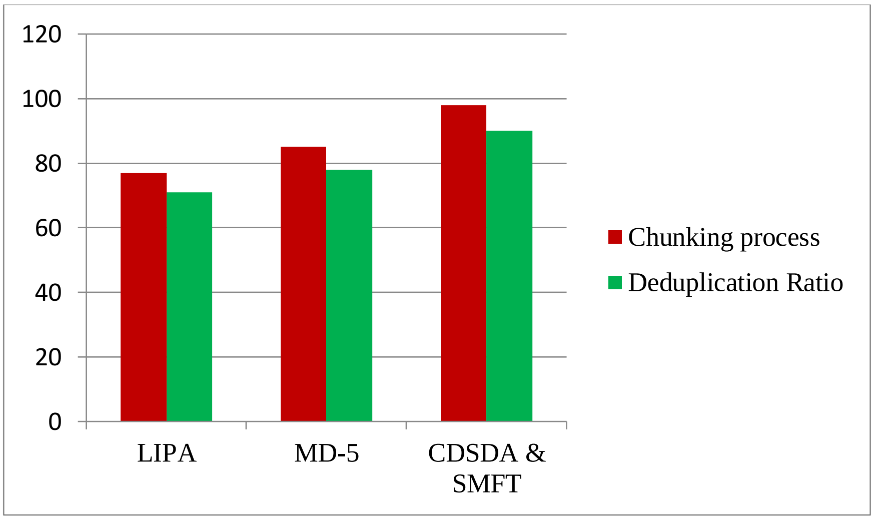
IV. RESULT AND DISCUSSION

In fig 3 shows that the comparison of the Machine Learning algorithm for generating chunking and deduplication analysis form large file time shows the comparison of the proposed method. The data deduplication analyzed through the chunk method can reduce system overload on the cloud system.

ALGORITHM	NUMBER OF CHUNKS	DEDUPLICATION RATIO	TIME IN SECOND
LIPA	623961	1.78300	2301
MD-5	1542374	1.81176	591
Proposed Algorithm	644933	2.06845	468
Bloom Filter	960091	2.1986	3349

Table 1: Performance of Time based Comparison algorithm

The result compared with proposed and existing algorithms for accurate analysis process the proposed method of machine learning algorithms are Semi-Supervised Multi-Label Filtering Techniques (SMFT) and Chunk Data Streaming Deduplication Analysis are used to partitioned and detect the duplicated data with accuracy this compared with an existing algorithm of LIPA and MD-5.



.

Figure 3: Comparison of Accurate classification

Deduplication ratio

The deduplication analyzing ratio of presented and existing works is given in the figure. From the comparison chart, it clearly says that the proposed system CDSDA of the machine learning system can give a high percentage of duplicated analysis ratio in the process of deduplication compared to existing methods LIPA, Bloom Filter, and MD-5. Hence, an improved system can give an effective analysis of the target of duplicated detection.

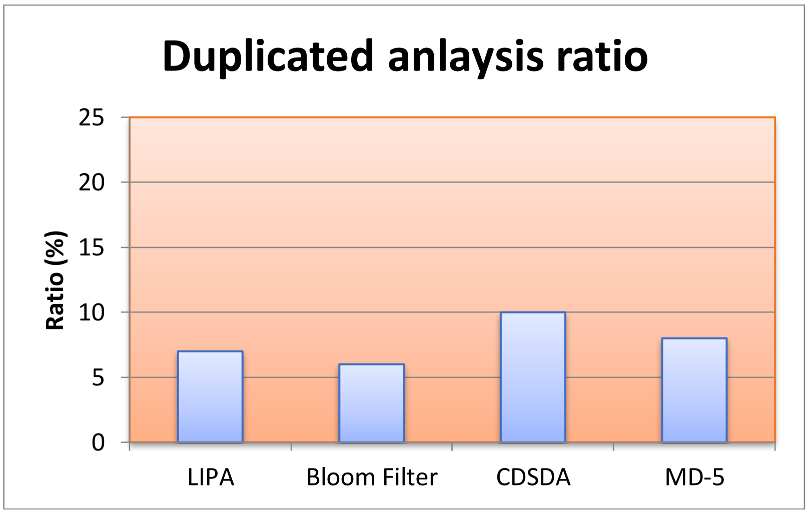


Figure 4: Data Deduplicated Ratio

V. CONCLUSION

Machine learning is intelligent for solve a problem with less time and efficiently. Due to the redundancy storing a large amount of data become complex to process the data and taking more time for query searching. The research concluded with two processes of chunk and deduplication using an ML algorithm for efficient storage. The first process used a Machine learning technique of SMFT algorithm to chunk the data by WFC, SC, and CDC partitioning the data into this type of format and store the unique key value into index terms of a repository. The second process to de-duplicate the data and efficiently store this research suggested a new technique to reduce the deduplication. Deduplication analysis by using the ML algorithm of CDSDA for removing the redundancy of data. The replication is analyzed by the comparison of index term storage. The partitioning of content is useful for efficient redundancy analysis and cloud storage.

REFERENCES

[1] ShubhanshiSinghal, AkankshaKaushik, and Pooja Sharma, ”A Novel Approach of Data Deduplication for Distributed Storage”, DOI: 10.14419/ijet.v7i2.4.10040 International Journal of Engineering & Technology, 7 (2.4) (2018) 46-52.

[2] Chi Yang and Jinjun Chen, "A Scalable Data Chunk Similarity-based Compression Approach for Efficient Big Sensing Data Processing on Cloud", DOI:10.1109/TKDE.2016.2531684, IEEE Transactions on Knowledge and Data Engineering.

[3] Thanh Trung Nguyen and Minh Hieu Nguyen, ”Distributed and High Performance Big-File Cloud Storage Based On Key - Value Store", International Journal of Networked and Distributed Computing, Vol. 4, No. 3 (July 2016), 159-172.

[4] Anju K S, Sadhik M S and Surekha Mariam Varghese, “Semantic Deduplication in Databases”, (IJITEE) ISSN: 2278-3075, Volume-8 Issue-6S, April 2019.

[5] GuangpingXu, Bo Tang and Hongli Lu, “LIPA: A Learning-based Indexing and Prefetching Approach for Data Deduplication”,2160-1968/19/$31.00 ©2019 IEEE DOI 10.1109/MSST.2019.00010.

[6] Moises G. de Carvalho, Marcos Andre Goncalves, and Altigran. S. da Silva, “A genetic programming approach to record deduplication”, IEEE Transactions on knowledge and data engineering, Vol.24, No. 3, March 2012.

[7] V. P. Archana Linnet Hailey and N. Sudha, “An Optimization Approach of Firefly Algorithm to Record Deduplication”, IJERT - ISSN: 2278-0181 Vol. 2 Issue 9, September – 2013.

[8] E. Ramirez, J. Brill, M.K. Ohlhausen, and J.D. Wright, "Data Brokers: A Call for Transparency and Accountability", Federal Trade Commission, May 2014.

[9] J. Li, J. Li, X. Chen and Z. Liu, “Privacy-preserving data utilization in hybrid clouds”, Future Generation Computer Systems, 30 (2014) 98-106.

[10] H. Dev, T. Sen, M. Basak, and M.E. Ali, "An approach to protect the privacy of cloud data from data mining based attacks", High Performance Computing, Networking, Storage and Analysis (SCC 2012), IEEE Computer Society, 2012, pp. 1106-1115.

[11] D. Sánchez, and M. Batet, “Toward sensitive document release with privacy guarantees”, Engineering Applications of Artificial Intelligence, 59 (2017) 23-34.

[12] Yinjin Fu, Hong Jiang, Nong Xiao, Lei Tian, and Fang Liu, "Application-Aware local global Source Deduplication for Cloud Backup Service of personal storage" IEEE International Conference on Cluster Computing in the Personal Computing Environment (2012).

[13] Easton, A., L. A. D. Webster, and M. J. Eacott. "The Episodic Nature of Episodic-Like Memories". Learning & Memory 19.4 (2012): 146-150. Web.

[14] D. Saxena, N. Kumar, and V. Singh, "A Cognitive Approach to solve Water Jugs Problem", International Journal of Computer Applications, vol. 124, no. 17, pp. 45-54, 2015.

[15] Nisha, T. R., S. Abirami, and E. Manohar, “Experimental study on chunking algorithms of data deduplication system on large scale data.”, In Proceedings of the International Conference on Soft Computing Systems, pp. 91-98. Springer India, 2016.

[16] H. Abdulsalam and A. Fahad, "Evaluation of Two Thresholds Two Divisor chunking algorithm using Rabin fingerprint, Adler, and SHA-1 hashing algorithms," The Iraqi Journal of Science, paper 4C.58, 2017.

[17] Periasamy J. K, and Latha B. “An enhanced secure content deduplication identification and prevention (ESCDIP) algorithm in cloud environment”, Neural Computing and Applications, 1-10. (2019).

[18] ZHU B, AND PATTERSON H, “Avoiding the disk bottleneck in the data domain deduplication file system”. In Proceedings of the 6th USENIX Conference on File and Storage Technologies 2008, pp. 18:1–18:14.

[19] Zhang, Yang, Wu, Yongwei and Yang, “Droplet: A distributed solution of data deduplication", IEEE/ACM Work Grid Comput. 2012;114–21.

[20] Gantz, John, and David Reinsel. “The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east.”, IDC iView: IDC Analyze future 2007.2012, pp 1-16, 2012.