International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8, Issue-1, May 2019

Comparsion of Time Overhead for Similarity Data Checking in Cloud Storage

S.Borgia Annie Catherine, S.Prasanna

Abstract*: *Cloud computing is a great technique to perform an enormous range and multipart computing of data. Cloud is an important platform for massive data storage. It is highly important to have the data quality for the effective storage and retrieval of data. Redundancy is a huge problem, which occurs due to the similar data getting stored in various content formats. there are a lot of techniques available shingle, Simhash, traits, TSA , PAS, EPAS to identify and remove duplicate but with the restriction in file control. the proposed technique AEPAS detects the data replication in a specific data compression method to eliminating duplicate copies of repeat data among many files with similarity content.

Index Terms*: *Similarity Detection, PAS, EPAS, Advanced EPAS..

I. INTRODUCTION

Cloud computing is an amazing innovation to execute a monstrous scale and complicated registering. It kills the necessity to keep up costly computing hardware, dedicated area, and software . Huge growth within the scale of data generated through cloud computing has been observed. Redundancy is a huge problem which occurs due to the similar data getting stored in various content formats. Cloud is an important platform for enormous data management, it is essential to manage data securely and efficiently. It is highly important to have the data quality for the effective storage and retrieval of data. Maintaining consistent data quality in the cloud can be achieved by means of similarity detection.

Similarity detection plays a very imperative job in information the executives. Identifying file similarity is an indispensable procedure. Sampling files is an effective approach to identify the file similarity algorithms such as Shingle, Simhash, Traits and Traditional Sampling Algorithm (TSA) are extensively used for detecting the similarities. But there are failures when identifying the similarities with these techniques.

Shingle, Simhash , Traits algorithms read the whole source document to figure the relating similarity character value, in this way requiring loads of CPU cycles and memory space and bringing about enormous plate gets to. Moreover, the overhead increments with the development of informational collection volume and results in a long delay. Rather than perusing the whole record, TSA tests a few information squares to compute the fingerprints as likeness attributes esteem. The overhead of TSA is fixed and negligible.

Position-Aware Sampling algorithm only identifies the file similarity in large data sets by modulo file length. PAS is effective when compared to Simhash algorithm, but still, there are near duplicates during the similarity detection. Enhanced Position-Aware Sampling algorithm concurrently samples data blocks and avoids the failure of similarity detection when shifting the bits or chunks. When the growth of sampling data is large then CPU Overhead increases and the efficiency is not obtained. EPAS still does not fully compare the content of the file. Hence an advanced technique is required to address the problem of EPAS. Advanced EPAS algorithm is proposed to detect and retrieve the similarity between the files in the large data set. This proposed algorithm is simple and efficient. When compared with EPAS, AEPAS has an improved metric and the CPU Overhead is completely low.

II. LITERATURE SURVEY

Anand Bhalera et al., [1] express the craft of the chunking algorithms utilized in data de-duplication process. Data de-duplication is one of the favorable methods used to decide distributed storage the executives issues. chunking algorithms are ordered into various classes dependent on spot, time and granularity. An alternate class of piecing calculations has numerous focal points and upgrades in distributed storage. AE is one of the promising piecing strategies. AE can be improved further by settling the piece estimate difference issue.

Xuandong An et al.,[2][9] acclimated a capable structure for packing thick point sets depicting a surface, the strategy chooses a subset of focuses, at that point it figures nearby portrayal of the chose focuses and uses the correspondence between the portrayal to encode them.

Hua et al. [3] investigated and abused data similarity which underpins proficient data position for the cloud. They structured a novel multi-center empowered and region delicate hashing that can precisely catch the distinction.

Manku et al. [4] utilized a Sim-Hash algorithm to identify similarity in web records having a place with a multi-billion page vault. Sim-Hash algoritm for all intents and purposes keeps running at Go.ogle web internet searcher consolidating with Google file framework.Andrei et al. [5], [6] anticipated a comparative website page discovery method called Shingle algorithm which abuses set activity to recognize similitude. Shingle is a common examining based methodology utilized to distinguish comparable website pages.

Revised Manuscript Received on 30 May 2019.
***** Correspondence Author

S.Borgia Annie Catherine*, Research Scholar, Department of Computer Application, VISTAS, Chennai, Tamilnadu, India.

DR.S.Prasanna, Department of Computer Application, VISTAS, Chennai, Tamilnadu, India.

So as to diminish the span of shingle, Andrei displayed Modem and Mins examining techniques. These algorithms are connected to the AltaVista web index at present.

Cox et al. [7] offered a similarity based system for finding a solitary source file, to play out a distributed reinforcement. They actualized a framework model called Pastiche.

Douglis et al. [8] used an alternate way to deal with handle the substance mindful de-duplication. In this strategy, the information is estimated as an article. Approaching data is changed over into the article and the equivalent has been contrasted and the as of now put away items for finding the copy in information successfully. Utilizing the Byte level examination and by the information of the substance of information, the information file is part into extensive information portions. The spat information fragments are compared with the as of now put away sections and comparative portions are resolved and adjusted. Toward the end the changed bytes are spared.

III. SIMILARITY TECHNIQUES

For identifying the data similarity, some of the related techniques are used. They are

1. Detection of similar web page with web search engine.
2. Detection of similarity using shingles in the storage systems.
3. Detection of similarity in the digital information which can be easily copied and retransmitted.
4. Detection of similarity via Remote file backup.
5. Detection of similarity using sensitive hashing.

Most of the above techniques has an increased CPU Overhead in case of large volume of data. Therefore, this research proposes an advanced EPAS algorithm AEPAS to detect the similarity and produce the results with less memory consumption and efficiency.

Following are some of the existing algorithms to detect similarity

- Shingle
- Simhash
- Traits
- PAS
- EPAS.

A. *SHINGLE*

The essential strategy for figuring likeness has two points of view:

First, similarity is communicated as a set crossing point issue.

Second, the overall size of crossing points is assessed by - a procedure of irregular sampling that should be conceivable unreservedly for each report. (The route toward assessing the general size of intersection purpose of sets can be connected to self-assertive sets).

Second, the overall size of crossing points is assessed by - a procedure of irregular sampling that should be conceivable unreservedly for each report. (The route toward assessing the general size of intersection purpose of sets can be connected to self-assertive sets).

B. *SIMHASH*

Data deduplication is a specialized data compression technique for eliminating duplicate copies of repeating data among many files. The intent of storage-based data deduplication is to inspect large volumes of data and identify large sections – such as entire file or large section of files – that are identical, in order to store only one copy of it.

Executing data deduplication

- Rabin carp fingerprinting
- MD5 hash of the established chunks
- Store the hashes and the file chunks into hash map with the MD5 hash as the key.
- Store the file as arrays of MD5 hashes.

C. *Traditional Sampling Algorithm (TSA)*

TSA does not peruse whole records, yet tests a few data squares to ascertain the fingerprints as similarity trademark esteems. TSA is simple and has a fixed overhead. Be that as it may, a slight modification will cause a disappointment of comparability distinguishing proof because of the moved piece positions.

D. *Position Aware Sampling Algorithm (PAS)*

PAS Algorithm comes into picture by solving the problem w.r.t shifting positions of sampling. PAS finds more genuine testing positions than that of TSA. The content of the sampling blocks have been moved because of the file modifications. Still PAS algorithm can maintain a strategic distance from the moving of testing positions created from slight document changes in the centre and the finish of source records.

Although PAS has an advantage over Simhash, it also has some disadvantages. As PAS have a place with I/O bound and CPU bound undertakings, ascertaining the Unicode of comparable documents requires loads of CPU relating cycles, the registering increments with the development of data sets. It normally requires a large amount of time to detect the similarity which results in long delays in case of larger data sets. Below example shows how the time consumption varies accordingly for the larger data sets.

To solve this problem, EPAS comes into picture. EPAS algorithm mainly focuses to reduce the resource consumption. EPAS will not read the entire file, instead its samples data blocks to squares to compute the unique mark as comparability trademark esteems, and revises the length to maintain a strategic distance from moved places of testing by modulo record length.

E. *Enhanced Position Aware Similarity Identification Algorithm (EPAS)*

EPAS constructs the data blocks into fingerprints by utilizing the hash functions.

A slight modification in the data blocks does not have an impact EPAS only finds the file based on the modulo length, the content of the file is not compared, so there is a high possibility of duplicate or redundant data getting stored in the cloud

EPAS maintains the advantages of both PAS and TSA. EPAS is an enhanced version of PAS. EPAS samples n/2 information blocks from the head and tail of the file which is modulated. EPAS maps these data blocks into finger prints by using hash function and obtains the similar characteristic value. Below is the EPAS Algorithm.

IV. PERFORMANCE ANALYSIS OF ADVANCED(EPAS)

AEPAS is proposed in order to address the problem in EPAS. AEPAS is linear and the sampling positions are cached with an advanced caching technique.

- A hash key is generated uniquely to store the sampling positions and the key is cached. As the keys are cached in an efficient way any sampling position can be retrieved randomly which is the major advantage of AEPAS. AEPAS has the same advantage of EPAS and PAS.
- Using 32-bit Hash function or 64-bit hash function relatively reduces the storage consumption of data blocks and identifies the similarity between the files effectively. AEPAS not only searches the file but also fetches the redundant data and stores it.
- Meanwhile, an improved metric is proposed to measure the similarity between different files and make the possible detection probability close to the actual probability.
- Below is the block diagram shows how the proposed system looks like.

TABLE 1: The Time overhead of Shingle, Simhash, Traits, PAS, EPAS and AEPAS.

	2 MB	5 MB	10 MB
Shingle	620	1300	3700
Simhash	530	538	552
Traits	472	478	476
PAS	72	71	69
EPAS	52	48	35
AEPAS	42	38	23

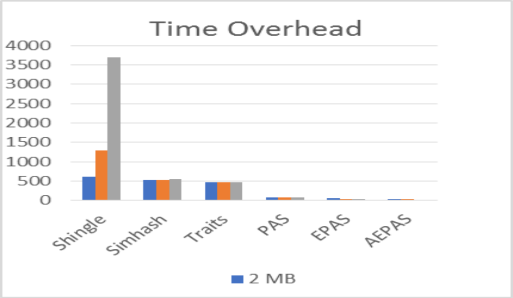

Fig. 2: The time overhead of AEPAS, EPAS, Shingle, Simhash, Traits and PAS algorithm with different file size 2 MB, 5 MB, and 10 MB

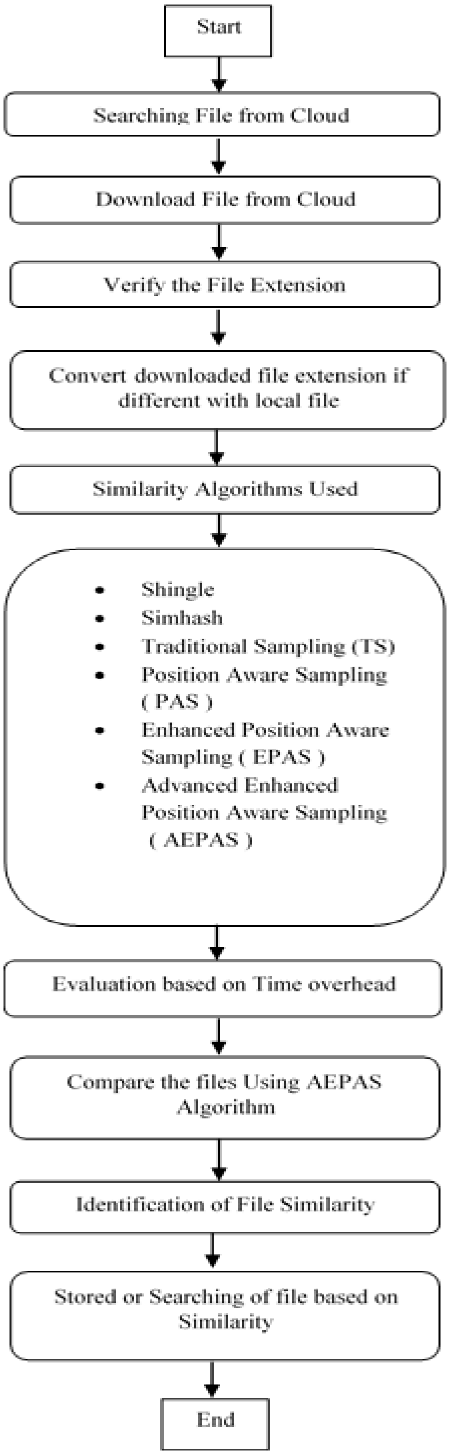
Start
Searching File from Cloud
Download File from Cloud
Verify the File Extension
Convert downloaded file extension if different with local file
Similarity Algorithms Used
Shingle
Simhash
Traditional Sampling (TS)
Position Aware Sampling (PAS)
Enhanced Position Aware Sampling (EPAS)
Advanced Enhanced Position Aware Sampling (AEPAS)
Evaluation based on Time overhead
Compare the files Using AEPAS Algorithm
Identification of File Similarity
Stored or Searching of file based on Similarity
End

Fig. 3. Flow of AEPAS, EPAS, Shingle, Simhash, Traits and PAS algorithm

V. CONCLUSION

The problem in cloud data storage is storing similar data with various aspects the proposed AEPAS is linear and sampling positions are cached with an advanced caching technique to identify the similarity among the data. In this paper, we executed an Advanced Enhance Position-Aware Sampling algorithm (AEPAS)to calculate time overhead for the data storage in a cloud environment. so Comprehensive tests are performed to choose Time overhead for AEPAS.

The optimal parameters(file content) used in AEPAS are not only its modulated file but also the file or the data chunk which is cached based on the generated unique id provided by the hash function. Due to this optimality, the similarity is determined efficiently with less time. Corresponding analysis and discussion of the Time overhead square measure introduced during this paper. The analysis of exactness and recall demonstrates that AEPAS is incredibly effective in detection file similarity in distinction to Shingle, Simhash, TSA, PAS, and EPAS. The experimental results conjointly recommend that the time overhead, of AEPAS are much less than that of those algorithms.

REFERENCES

1. Bhalerao, A., & Pawar, A. (2017, May). A survey: On data deduplication for efficiently utilizing cloud storage for big data backups. In 2017 International Conference on Trends in Electronics and Informatics (ICEI) (pp. 933-938). IEEE.
2. An Xuandong, Xiaogang Yu, and Yifan Zhang. "Research on the self-similarity of point cloud outline for accurate compression." (2015): 5-5.
3. Y. Hua, X. Liu, and D. Feng, "Data similarity-aware computation infrastructure for the cloud," IEEE Transactions on Computers, p. 1, 2013
4. M. S. Charikar, "Similarity estimation techniques from rounding algorithms," in Proceedings of the thirty-fourth annual ACM symposium on Theory of computing. ACM, 2002, pp. 380–388
5. A. Z. Broder, "On the resemblance and containment of documents," in Compression and Complexity of Sequences 1997. Proceedings. IEEE, 1997, pp. 21–29.
6. A. Z. Broder, S. C. Glassman, M. S. Manasse, and G. Zweig, "Syntactic clustering of the web," Computer Networks and ISDN Systems, vol. 29, no. 8, pp. 1157–1166, 1997.
7. L. P. Cox, C. D. Murray, and B. D. Noble, "Pastiche: Making backup cheap and easy," ACM SIGOPS Operating Systems Review, vol. 36, no. SI, pp. 285–298, 2002.
8. L. P. Cox, C. D. Murray, and B. D. Noble, "Pastiche: Making backup cheap and easy," ACM SIGOPS Operating Systems Review, vol. 36, no. SI, pp. 285–298, 2002.
9. Dharmarajan, K., and M. A. Dorairangaswamy. "Web User Navigation Pattern Behavior Prediction Using Nearest Neighbor Interchange from Weblog Data." International Journal of Pure and Applied Mathematics 116.21 (2017): 761-775.