A Concise Study on Data Duplication Removal using File Checksum

Debashree Sadagar
MCA Scholar
School of CS and IT, Dept of MCA
Jain Deemed-to-Be-University,
560069
Bangalore, India
19mcar0070@jainuniversity.ac.in

Dr. Mir Aadil
Asst. Professor
School of CS and IT, Dept of MCA
Jain Deemed-to-Be-University, 560069
Bangalore, India
mir.aadil@inurture.co.in

Abstract- The goal of the project is to develop software that uses file checksums to prevent data duplication. Notepad++ and Visual WAMP Server were used to create the project. It is used to design and develop project software. My SQL was used to construct and maintain all databases.
The project's main goal is to reduce the amount of duplicates in one type of NoSQL database, particularly the key-value store, to improve process performance so that the backup window is not impacted, and to design for horizontal scaling so that it can compete on a Cloud Platform.
Keywords: Database, Duplication, Entity, Data, Checksum, Redundant, User id.

1. Introduction

The globe generates a massive amount of digital data, which is continually growing. Rapidly growing data creates a slew of problems for current storage systems. Extra storage space is required when there is a large amount of data. More data storage is required for backup as the amount of data grows.
Data de-duplication is a specialized data compression technology that eliminates redundant data and improves storage efficiency. Duplicate data is eliminated and not maintained during the de-duplication process, leaving a unique copy of the data chunk to be stored and a pointer to that copy. De-duplication is a method of lowering storage requirements by only storing data that is unique.
Data de-duplication can save anywhere from ten to fifty times the amount of data. Storage costs are reduced since smaller drives and fewer disc purchases are required with less storage. Because there is less data, there are fewer backups, which means shorter backup windows and faster recovery times. As a result, data de-duplication technology allows it to offer a higher number of users with the same amount of storage capacity.
In data duplication technologies, the file checksum technique is widely used to quickly and precisely identify redundant data. A checksum can be used to determine whether data is redundant or not. However, there are times when false positives occur. We must compare a fresh chunk of data with previously stored chunks of data to avoid false positives. To reduce the time, it takes to rule out false positives, the current study used file data checksum extraction. The user id, filename, size, extension, checksum, and date-time table are all stored in the target file, whereas the user id, filename, size, extension, checksum, and date-time table are all stored in the source file. When a user uploads a file, the system first generates a checksum, which is then compared to the database's checksum data.

The item will be updated if the file already exists; else, a new record will be generated

in the database. The database will be stored in the cloud, and a link between the application and the cloud server will be formed. Data de-duplication is crucial for minimizing storage usage and making it more manageable amid today's massive data growth. The project's main goals are to eliminate duplication in one type of NoSQL database, the key-value store, to improve process performance to the point that the backup window is only marginally affected, and to design with horizontal scaling in mind so that it can compete on a Cloud Platform. The project will be accessed via a link in a web browser because the project files and database file will be kept in the cloud.

2. Literature Survey

Many duplicate files of significant size will accumulate on the computer when maintaining and conducting file operations on it or on other storage media. The accumulation of this digital garbage might lead to a lack of storage space and a decline in computer performance. As a result, it's necessary to search for and delete duplicate files from your computer's hard disc. It is sometimes useful to have information about files that have several replicates. If your computer has multiple copies of a requested file, they will all be stored in RAM, slowing down your system.

We employ a data deduplication technique in which the system checks the checksum every time a file is uploaded, and the checksum verifies checksum information kept in the database. If the file is already there, the section shall be refreshed; otherwise, a new entry will be made into the database. The Duplicate File Detector and Remover will assist you in reclaiming critical disc space and increasing data efficiency. Duplicates can be deleted to assist speed up indexing and minimize backup time and size. It can swiftly and safely locate any undesirable duplicate files on the system, delete them, or relocate them to a different location, depending on the user's preferences. Your system will be freed of duplicates.

Data deduplication tools check data down to the block and bit level, and only the altered data is saved after the initial occurrence. The rest is discarded, and a pointer to previously saved data is substituted. Under the correct conditions, block- and bit-level deduplication algorithms can achieve compression ratios of 20x to 60x, or even more.

Different de-duplication techniques are investigated in this work, where de-duplication tactics are performed on encrypted data in a large storage area. The majority of the solutions discussed here are based on Confluent encryption, which is a basic way for deduplication that works well with encoded data.

According to the paper's authors, a methodology should be devised that will increase storage capacity without consulting an encryption strategy, by using a de-duplication system in data or information storage servers that scrambles the available data.[1]

In this paper, we looked at modern-day backup systems and applied an unique technique to help reduce fragmentation. The leading backup tool has sections of each backup that are presumably bodily scattered, causing a time-consuming fragmentation issue. Fragmentation promotes the upward movement of two types of fragmented containers: sparse and out-of-order boxes. On both the restore and garbage series, the device's sparse container exacerbates the device's average overall performance. During a repair, out-of-order packing containers interact with those that can be accessed often. The authors use an age-old set of guidelines and a stockpile-aware filter out to reduce fragmentation. Those assist in determining the scant and out of order discipline.[2]

This paper developed a secure deduplication solution based on convergent and MECC algorithms in a cloud-fog scenario. The proposed method is tested in four different scenarios: a) when new users

try to upload a new file, b) when the same user tries to post the same file, c) when several users try to upload the same file, and d) when many users try to download the file. The proposed system's performance was tested with files ranging in size from 5 MB to 25 MB, with each iteration rising by 5 MB. The new system has a security rating of 96 percent, which is a promising result and higher than the other encryption methods already in use, according to the performance analysis.[3]

To ensure successful data processing and analysis, addressing rising storage needs is a difficult and time-consuming effort that necessitates a big computational infrastructure. Data deduplication is becoming increasingly important for cloud storage providers as the amount of customers and the amount of their data grows at an exponential rate. Cloud providers save a lot of money on storage and data transfer by storing a unique copy of duplicate data. Cloud computing, cloud file services, accessibility, and storage are all included in this project. It also examines existing data de-duplication strategies, procedures, and implementations for the benefit of cloud service providers and cloud users. The project also presents an efficient way for finding and eliminating duplicates using file checksum algorithms by calculating the digest of files.[4]

Cloud service providers are enticed to utilize data de-duplication to save money on storage and bandwidth. Cloud users want to be able to use the cloud safely and discreetly in order to preserve the data they share there. As a result, before transferring data to the cloud, they encrypt it. The data deduplication feature becomes a difficult problem because the encryption goal competes with the de-duplication function. In terms of security and efficiency, existing de-duplication solutions are insecure and inefficient. They are either computationally expensive or vulnerable to brute force assaults, which allow the attacker to access files. This is what drives the authors to suggest an efficient and secure method for eliminating duplicate data.

The authors begin by describing the implementations and functionality of de-duplication techniques, then go on to a review of literature that suggests several ways to duplication removal as well as the security and efficiency issues that current systems face. The authors have recommended using the AES-CBC algorithm and hashing algorithms to boost the effectiveness and security of data de-duplication for users. Without the involvement of a third party, users' keys are created in a consistent and secure manner. The authors illustrate the efficacy of the recommended approach by putting it into practice and comparing it to existing techniques.[5]

The approval of data de-duplication is addressed in this work. While data de-duplication is similar to classic de-duplication, it considers a variety of customer benefits. Engineers might be able to support more creative de-duplication development solutions if there are less new copy checks introduced. According to security investigations, the procedure is secure when compared to the descriptions in the projected security model.

The goal of this research is to develop a prototype of a recommended, sanctioned copy check plan and test it. The study will demonstrate that the proposed prototype is accountable for minor overhead-differentiated archetypal processes.[6]

The suggested technique employs a Content-Based chunking algorithm with Variable Chunking use, which is implemented using Rabin Karp Rolling Hash (RKRH). RKRH is a data chunking method for breaking down huge files into smaller bits. In principle, the proposed technique works by computing a data piece's hash value, which may be compared to a fingerprint. The chunk availability technique is then used to determine whether this chunk exists in the storage; if it does not, a reference to it is added, and the hash value is saved as a key in the storage. The

suggested solution employs data chunk compression to eliminate data redundancy within the same chunk.

In effect, the proposed method achieves a data de-duplication ratio of 33% and an average upload lag of five seconds. Finally, the proposed approach can be utilized with any data file type as a byte stream.[7]

The authors of this paper remove duplicate data to save storage space and boost network storage speed. To discover duplicate data in a cloud setting, the authors used the inverted index approach and tf-idf. After data duplication is complete, the system is designed for secure data transformation in the network.

Cryptography is a typical method for safeguarding data in the cloud. In an information security system, the encryption algorithm is crucial. Data encryption and decryption are used to achieve security. The authors of this research look at a secure deduplication technique. Markle hash tree is used after duplicate data is removed.[8]

To address the issues of poor effect and low efficiency of traditional information de-duplication approaches, this research proposes a rapid de-duplication approach for text backup information in library databases using big data. To begin, this paper performs parallel mining of text information features in a library database, determining the parameter value of the repeated feature function using features with strong classification ability, obtaining entries with the parameter value higher than the threshold value, determining the number of text repeated backup information and the group weight, setting the difference between the two as the remaining digits, and stopping de-duplication when the rem value is higher than the threshold value,

The results of the trial show that this method has a 96.95 percent average accuracy, a weight removal efficiency of always better than 98 percent, and a proper weight removal effect. [9]

Data deduplication, sometimes referred to as "Dedup," is a technique for reducing the impact of redundant data on volume expenses. By examining each byte in each packet, repetition disposal or deduplication over network packets necessitates massive registering assets to locate fundamental units of rehashed substance, known as pieces. Deduplication expands capacity significantly, especially when applied to large amounts of data. As a result, this article explored a variety of data deduplication types and methodologies, as well as the importance of client-side deduplication and a comparison of a few of the existing client-side deduplication schemes. [10]

3. Conclusion

This technique focuses on creating a web-based tool that can rapidly and simply discover redundant data using the file checksum technique. The Message Digest (MD-5) technique is used to calculate the checksum of both existing and new files. The MD-5 technique is used to determine the checksum as well as providing improved security and encryption for users' valuable files. As a result, by providing superior security, this system easily and rapidly removes duplicate files.

Businesses that deal with highly redundant activities that require regular data copying and storage for future reference or recovery can benefit from the web application. The method can be used in backup and disaster recovery solutions because it allows organizations to save data often and encourages speedy, reliable, and cost-effective data recovery. A file backed up once a week, for example, generates a lot of duplicate data and takes up a lot of disc space. File checksum data duplicate elimination performs an analysis and removes these sets of duplicate data, leaving only the unique and essential data, resulting in significant storage space savings. The biggest stumbling block was that none of the database's files should be duplicates of one another.

REERENCES

[1]S.Usharani, K.Dhanalakshmi, N.Dhanalakshmi, "De-Duplication Techniques: A Study" International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-7, Issue-6S5, April 2019.
[2] V.Sathiya Suntharam, Sheo Kumar, Chandu Ravi Kumar, "Research Method of Data Deduplication Backup System" International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8, Issue-11S2, September 2019.
[3] Shynu P.G, Nadesh R.K, Varun G. Menon, Venu P., Mahdi Abbasi & Mohammad R. Khosravi, "A secure data deduplication system for integrated cloud-edge networks" Journal of Cloud Computing 9, article number 61(2020).
[4] Osuolalea.Festus, "Data Finding, Sharing and Duplication Removal in the Cloud Using File Checksum Algorithm" International Journal of Research Studies in Computer Science and Engineering (IJRSCSE), Volume 6, Issue 1, 2019, PP 26-47, ISSN 2349-4840 (Print) & ISSN 2349-4859 (Online).
[5] Nourah Almrezeq , Mamoona Humayun , A. A. Abd El-Aziz,and NZ Jhanjhi "An Enhanced Approach to Improve the Security and Performance for Deduplication",Turkish Journal of Computer and Mathematics Education , 2866 Research Article Vol.12 No.6 (2021), 2866-2882.
[6] Korre, Mahender, "Security and Data De-Duplication Using Hybrid Cloud Technology" (2017). Culminating Projects in Information Assurance. 22.
[7] Samer 0 Majed, Sawsan K. Thamer, "Cloud Based Industrial File Handling and Duplication Removal Using Source Based Deduplication Technique", AIP Conference Proceedings 2292, 030012 (2020).
[8] Ajahar Ismailkha Pathan, Liladhar M. Kuwar, Rijavan A. Shaikh, Dheeraj Basant Shukla, "Removing Duplicate Data in Cloud Environment using Secure Inverted Index Method", International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056, Volume-05 Issue: 09, Sep 2018.
[9] Ling Ji, "Research on Fast De-Duplication of Text Backup Information in Library Database Based on Big Data", International Journal of Information and Communication Technology, 2021 Vol.19 No.1, pp.76 – 92.
[10] Priya J, Vinothini C, Dinesh P S, Reshmi T S, "Data Deduplication Techniques: A Comparative Analysis", International Journal of Aquatic Science ISSN: 2008-8019 Vol 12, Issue 03, 2021.