International Journal of Engineering and Techniques - Volume 4 Issue 2, Mar-Apr 2018

RESEARCH ARTICLE OPEN ACCESS

AVOIDING DUPLICATION DATA IN HDFS BASED ON SUPERVISED LEARNING

ALAPATI JANARDHANA RAO [*1], KOPPOLU SREE VENKATA GOPI NAGA RAJU[2]

Abstract:

The Hadoop Distributed File System (HDFS) part of Apache Hadoop helps in conveyed capacity of huge information with a group of item equipment. HDFS guarantees accessibility of information by duplicating information to various hubs. Be that as it may, the replication strategy of HDFS does not think about the notoriety of information. The prevalence of the documents tend to change after some time. Thus, keeping up a settled replication factor will influence the capacity effectiveness of HDFS. In this paper we propose a proficient dynamic information replication administration framework, which consider the ubiquity of documents put away in HDFS before replication. This methodology powerfully characterizes the records to hot information or cool information in view of its prominence and builds the reproduction of hot information by applying eradication coding for icy information. The trial comes about demonstrate that the proposed technique viably decreases the capacity usage up to 40% without influencing the accessibility and adaptation to internal failure in HDFS.

Keywords: Big Data, Hadoop Distributed File System, Dynamic data replication.

1. INTRODUCTION

A versatile replication administration (ARM) framework is intended to give high accessibility to the information in HDFS by means of improving the information area metric. Accordingly, the profoundly nearby accessible information enhances the execution of the Hadoop framework. It is important that the eradication code is connected to keep up the unwavering quality. A multifaceted nature diminishment strategy for the forecast procedure is proposed in both hyper-parameter learning and preparing stages. This proposed technique altogether expands the execution as far as response rate for the replication procedure while as yet keeping the precision of the prediction.ARM in HDFS is actualized here and an assessment is done to basically check the viability of the proposed strategy as contrasted and the cutting edge strategy. Enormous information has turned into the new wilderness of data administration given the measure of information the present frameworks are creating and expending. It has driven the requirement for mechanical framework and devices that can catch, store, break down and envision immense measures of dissimilar organized and unstructured information. This information are being created at expanding volumes from information serious advancements including, yet not restricted to, the utilization of the Internet for exercises, for example, gets to data, person to person communication, portable processing and business. Companies and governments have started to perceive that there are unexploited chances to enhance their ventures that can be found from these information. Most corporate ventures confront critical difficulties in completely utilizing their information. Every now and again, information is secured away different databases and preparing frameworks all through the undertaking, and the inquiries clients and examiners solicit require a total view from all information, here and there totalling many terabytes.

2. LITERATURE SURVEY

Q. Wei, B. Veeravalli, B. Gong, L. Zeng, and D. Feng

Information replication has been generally utilized as a mean of expanding the information accessibility of extensive scale distributed storage frameworks where disappointments are ordinary. Planning to give financially savvy accessibility, and enhance execution and load-

adjusting of distributed storage, this paper displays a practical dynamic replication administration conspire alluded to as CDRM. A novel model is proposed to catch the connection amongst accessibility and imitation number. CDRM use this model to compute and keep up negligible imitation number for a given accessibility necessity. Copy arrangement depends on limit and blocking likelihood of information hubs. By altering reproduction number and area as indicated by workload changing and hub limit, CDRM can progressively redistribute workloads among information hubs in the heterogeneous cloud. We actualized CDRM in Hadoop Distributed File System (HDFS) and test comes about decisively show that our CDRM is savvy and beats default replication administration of HDFS as far as execution and load adjusting for vast scale distributed storage.

G. Ananthanarayanan, S. Agarwal, S. Kandula, A. Greenberg, I. Stoica, D. Harlan, and E. Harris

To enhance information accessibility and strength MapReduce structures utilize record frameworks that imitate information consistently. Notwithstanding, investigation of employment logs from a substantial generation bunch indicates wide dissimilarity in information prominence. Machines and racks putting away mainstream content move toward becoming bottlenecks; along these lines expanding the finishing times of occupations getting to this information notwithstanding when there are machines with save cycles in the group. To address this issue, we exhibit Scarlett, a framework that repeats squares in light of their notoriety. By precisely foreseeing document ubiquity and working inside hard limits on extra stockpiling, Scarlett makes negligible impedance running employments. Follow driven reenactments and investigations in two well known MapReduce structures (Hadoop, Dryad) demonstrate that Scarlett successfully eases hotspots and can accelerate occupations by 20.2%.

K. S. Esmaili, L. Pamies-Juarez, and A. Datta

Eradication codes are a necessary piece of numerous circulated stockpiling frameworks went for Big Data, since they give high adaptation to internal failure to low overheads. Nonetheless, conventional deletion codes are wasteful on perusing put away information in corrupted situations (when hubs may be inaccessible), and on renewing lost information (indispensable for long haul flexibility). Therefore, novel codes streamlined to adapt to disseminated capacity framework subtleties are energetically being looked into. In this paper, we take a designing option, investigating the utilization of basic and develop procedures - comparing a standard deletion code with RAID-4 like equality. We do a logical examination to decide the adequacy of this approach over conventional and some novel codes. We expand upon this investigation to configuration CORE, a general stockpiling crude that we coordinate into HDFS. We benchmark this usage in a restrictive bunch and in EC2. Our tests demonstrate that contrasted with customary deletion codes, CORE utilizes half less transmission capacity and is up to 75% speedier while recouping a solitary fizzled hub, while the additions are separately 15% and 60% for twofold hub disappointments. Cerri et al proposed Knowledge in the cloud set up of information in the Cloud to help synergistic assignments which are computationally serious and encourage dispersed, heterogeneous learning. This is named as Utility Computing got from required information all through Cloud the utilities like power, gas for which we pay for what we use from a common asset. With the developing enthusiasm for cloud, examination is a testing errand. When all is said in done, Business Intelligence applications, for example, picture preparing, web seeks, understanding clients and their purchasing propensities, supply chains and positioning and Bio-informatics (e.g. quality structure expectation) are information escalated applications. Cloud can be an ideal counterpart for taking care of such expository administrations. For instance, Google's MapReduce can be utilized for investigation as it shrewdly lumps the information into littler capacity units and disseminates the calculation among minimal effort handling units. A few research groups have begun chipping away at making Analytic systems and motors which enable them to give Analytics as a Service. For instance, Zementis propelled the ADAPA prescient investigation choice motor on Amazon EC2, enabling its clients to send, incorporate, and execute factual scoring models like neural systems, bolster vector machine (SVM), choice tree, and different relapse models.

Wei et al. (2010) [5] proposed a cost-effective dynamic replication strategy for the cloud storage systems which is referred as CDRM. In this work the popularity of a data file is calculated to create replica for the data file. After finding the popular file the replica is placed in a suitable node considering the blocking probability and capacity of the nodes. This method concentrates on capturing the relationship between availability and replica number. They do not consider the availability of files that has low replica factors.

Ananthanarayanan et al. (2011) [6] proposed an off-line system called Scarlett, which periodically replicates popular files using prediction method based on the historical usage and jobs submitted for execution. It also distributes the replicas among clusters with the goal of minimizing hotspots. This method uses the concept of aging for replicas to give space for new replicas. This may cause loss of some files.

Abad et al. (2011) [7] proposed an adaptive data replication for efficient cluster scheduling (DARE), which replicates the data files dynamically to increase data locality. The number of replicas to be created for each file and the node to place the replica is determined based on the probabilistic sampling and a competitive aging algorithm independently at each node. The replication decision is based on probability and does not consider the trends of data utilization.

Kaushik et al. (2011) [8] proposed a predictive data replication policy for GreenHDFS which proactively create and delete replicas based on the file heat predictions. The file's heat is obtained based on the total number of access to the file and the file's hot lifespan. Replicas are created for the hot files and the replica of cold file gets deleted. This method fails to consider the management of cold data efficiently and the directory structure is an important factor in the anticipation for file accesses.

M. Bsoul et al. (2011) [9] proposed a replication strategy for data grids which considers the factors like frequency of requests, size of file, etc for replicating the data. This strategy does not consider the scenario of varying user behavior.

Cheng et al. (2012) [10] proposed an elastic replica management system (ERMS), which uses an active/standby model for the storage of data in HDFS. The data is classified into hot or cold by using the complex event processing engine and replica is created dynamically based on this classification. Erasure code is applied to unpopular data to save the storage space.

Kousiouris et al. (2013) [11] proposed a proactive data management in Hadoop clusters which is based on predictive data activity patterns. The method will predict the future data demand in the Hadoop cluster using Fourier series analysis [12]. In this method file is classified only in limited replication scenarios.

Bui et al. (2016) [13] proposed an adaptive replication management in HDFS based on supervised learning. This method replicate the data files based on the predictive analysis.The popularity of each data file is predicted using probability theory and replicate the high potential files. Erasure coding is applied to low potential files to ensure reliability.

Qu et al. (2016) [14] proposed a dynamic replication strategy (DRS) based on Markov Model for HDFS. In this method a transition probability matrix is constructed based on the accessing of files over time and then calculates the stationary probability distribution of the system. Using the results obtained data is classified as hot or cold. Extra replica is created for hot data and replica of cold data is deleted. This method is not considering the effective management of cold data resulting in a probability of data loss.

3. BACKGROUND THEORY

The Hadoop framework plays an important role in the handling and processing of big data. Two main components of Hadoop are MapReduce and Hadoop Distributed File System (HDFS). The MapReduce is an algorithm which helps in the processing of large data by implementing parallel processing. MapReduce consist of two parts, a Map task and Reduce task. These two tasks combined together perform the processing tasks. The HDFS is the distributed storage system which handles the storage of files in Hadoop. HDFS provides a reliable and fault tolerant architecture to store files. It follows a rack

based clustering, in which nodes are stored in racks and a cluster is formed combining these racks. A file entered to the HDFS is divided into blocks of equal size except the last block. These blocks are replicated and stored in separate nodes. By default HDFS creates three copies for each block. HDFS manages the placement of these replicas in such a way that two blocks are stored in the same rack and one in a separate rack. By following this method HDFS ensures the availability of a block even if a node fails or even if an entire rack goes down.

The operations in HDFS consist of two types of nodes Data Nodes and NameNode. Namenode manages the operation in the HDFS cluster. Datanodes are the nodes in which the blocks are stored. Namenode holds the information about each data node in the cluster and the details of blocks stored in it. When a client wants to read or write to HDFS, it first communicates with the namenode and the namenode provides the information regarding the blocks and nodes. After acquiring these information the client communicates directly to the datanode for reading or writing. The operations in HDFS consist of two types of nodes Data Nodes and NameNode. Namenode manages the operation in the HDFS cluster. Datanodes are the nodes in which the blocks are stored. Namenode holds the information about each datanode in the cluster and the details of blocks stored in it. When a client wants to read or write to HDFS, it first communicates with the name node and the namenode provides the information regarding the blocks and nodes. After acquiring these information the client communicates directly to the datanode for reading or writing.

Erasure coding (EC)[15] is a method of data protection in which data is broken into fragments, expanded and encoded with redundant data pieces and stored across a set of different locations on storage media [4]. Reed-Solomon (RS) encoding is a popular erasure coding method. In this method the data is divided into equal blocks and a set of parity blocks are added to it. So even if some of the blocks went missing the original file can be recreated with the help of these parity blocks. Reed-Solomon (10,4) configuration splits the file into 10 blocks consisting of 6 data blocks and 4 parity blocks. The proposed work combines the erasure coding with the files in HDFS to provide fault tolerance and availability. Reed-Solomon (10, 4) erasure coding is used in this work, since it can survive four block failures. This assures the prevention of data loss while cold data is stored as erasure coded file.

4. PROPOSED SYSTEM

Hypothetically, by setting the potential reproductions on low usage hubs (low blocking rate hubs), the replication administration diverts the undertakings to these sit still hubs and adjust the calculation. The blocking rate is ascertained in light of the data gave by the observing framework. In light of Ganglia structure, the checking framework is straightforward, strong and simple to arrange for observing the vast majority of the required measurements. Subsequent to connecting to the HDFS hubs, the checking framework can gather measurements by means of Ganglia API.

- We composed a versatile replication administration (ARM) framework to give high accessibility to the information in HDFS by means of improving the information region metric. Subsequently, the profoundly neighborhood accessible information enhances the execution of the Hadoop framework. It is significant that the eradication code is connected to keep up the unwavering quality.
- We proposed an unpredictability diminishment strategy for the forecast method in both hyper-parameter learning and preparing stages. This proposed technique altogether builds the execution as far as response rate for the replication methodology while as yet keeping the exactness of the forecast.
- We actualized ARM in HDFS and did an assessment keeping in mind the end goal to for all intents and purposes check the adequacy of the proposed strategy as contrasted and the best in class technique.

ADVANTAGES OF PROPOSED SYSTEM:

- The fundamental capacity of the proposed architecture is to powerfully scale the replication factors and in addition to proficiently plan the position of copies in light of the entrance capability of every datum record.
- Additionally, to diminish the figuring time, the information base and heuristic strategy are actualized to distinguish the likeness in the entrance design between in-handling records and the anticipated ones.

- By definition, the entrance design is really a set of eigenvectors portraying the component properties of prepared information.
- Two records with comparative access practices are treated with a similar replication methodology. In any case, in light of the fact that these strategies are minorparts and famously utilized as a part of different frameworks, examining them isn't inside the extent of this paper.

Proposed Dynamic Data Replication Algorithm

Input: log
Begin
1. Set time interval
2. For each time interval
{
i. read logfile
ii. for each file f_i
{
a. Find ac_i, nc_i, rf_i
b. Calculate popularity index(PI_i) of each file

$$PI_i = (ac_i * nc_i) / rf_i$$

}
iii. Calculate the threshold,

$$T = \frac{\sum_{i=1}^{n} PI_i}{n}$$

iv. For each file f_i
Compare threshold T
If $PI_i >= T$
hd ← f_i
Else
cd ← f_i
v. For each f_i in hd,
Increment rf_i by 1.
vi. For each f_i in cd
Set rf_i to1
Encode f_i using Reed-Solomon erasure code
}end for
End

5. RELATED WORK

In HDFS, so as to guarantee information accessibility and to diminish the possibility of information misfortune, each document is duplicated over various machines. The default replication factor in HDFS is to make three copies for each record. HDFS replication methodology won't consider whether a specific document is well known or not. Pointless replication of non-mainstream record will bring about capacity overhead. In the proposed procedure, a dynamic information replication calculation is utilized to deal with the imitations in HDFS. The Replication Management System in the proposed calculation deals with the replication of documents in HDFS. This module arranges the information records into hot information or accessibility. Replication Management System does these errands with the assistance of HDFS Logging System. The logging framework give subtle elements, for example, the quantity of records got to, their source, the hubs which got to them, recurrence of access for each document, and so on. The Logging framework gets
all these data from HDFS and gives it to the Replication administration System. a Hadoop group was setup
including 10 hubs. The physical Hadoop group involves one ace hub and nine slave hubs and the adaptation of a Hadoop dissemination is 2.7. The ace hub goes about as both name hub and information hub, accordingly a
bunch of ten information hubs is shaped. Every hub is furnished with Intel Core i5 (3.30GHz) CPU and 8 GB RAM. Documents were replicated into HDFS from the nearby record framework.

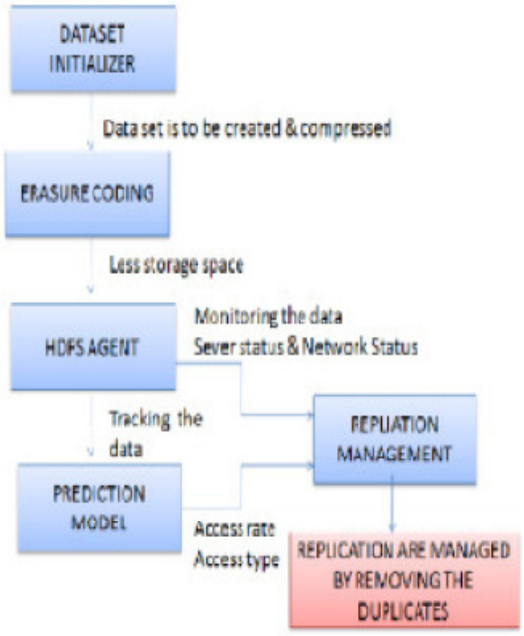

Fig:-Block Diagram of HDFS

Records of various size and sorts are considered for the experimentation of the calculation. Different sorts of records including content, sound, video documents with sizes going from the 600MB to 4GB were considered for this reason. The records were partitioned into obstructs by HDFS with default square size of 128MB. At first, the replications for the documents were three, which is

the default replication tally in HDFS. These documents were gotten to arbitrarily from various hubs, at various time interims. The log documents were examined. At that point in normal interims the calculation is executed in the HDFS bunch. The calculation checks the entrance mean each document and computes their prevalence. In view of this prominence esteem and edge, documents are ordered into two hot or cool. Replication mean hot records is increased and icy documents are encoded utilizing Reed-Solomon deletion code. The execution of the calculation was dissected looking at the outcome got by utilizing Hadoop default replication technique.

6. CONCLUSION

In this paper, to upgrade the availability of HDFS by enhancing the data region, our dedication revolves around following core interests. At first, we diagram the replication organization structure which is truly adaptable to the typical for the data get the chance to outline. The approach not simply master successfully plays out the replication judiciously, yet also, keeps up the faithful quality by applying the erasure coding approach. Second, we propose a multifaceted nature diminishment technique to disentangle the execution issue of the normal framework. Frankly, this multifaceted nature diminishment technique on a very basic level revives the desire methodology of the passageway potential estimation. Finally, we execute our technique on a honest to goodness gathering and affirm the practicality of the proposed approach. With a careful examination on the characteristics of the report exercises in HDFS, our uniqueness is to make an adaptable response for moving the Hadoop structure.

REFERENCES

[1] The Gartner website [Online]. Available: http://www.gartner.com/it-glossary/big-data/.

[2] The Apache Hadoop website. [Online]. http://hadoop.apache.org/.

[3] The Apache Hadoop Guide website. [Online]. http://hadoop.apache.org/common/docs/stable/hdfs_design.html.

[4] Shvachko, K., Kuang, H., Radia, S., & Chansler, R. (2010, May). The hadoop distributed file system. In Mass storage systems and technologies (MSST), 2010 IEEE 26th symposium on (pp. 1-10). IEEE.

[5] Wei, Q., Veeravalli, B., Gong, B., Zeng, L., & Feng, D. (2010, September). CDRM: A cost-effective dynamic replication management scheme for cloud storage cluster. In Cluster Computing (CLUSTER), 2010 IEEE International Conference on (pp. 188-196). IEEE.

[6] Ananthanarayanan, G., Agarwal, S., Kandula, S., Greenberg, A., Stoica, I., Harlan, D., & Harris, E. (2011, April). Scarlett: coping with skewed content popularity in mapreduce clusters. In Proceedings of the sixth conference on Computer systems (pp. 287-300). ACM

[7] Abad, C. L., Lu, Y., & Campbell, R. H. (2011, September). DARE: Adaptive data replication for efficient cluster scheduling. In Cluster Computing (CLUSTER), 2011 IEEE International Conference on (pp. 159-168). IEEE

[8] Kaushik, R. T., Abdelzaher, T., Egashira, R., & Nahrstedt, K. (2011, July). Predictive data and energy management in GreenHDFS. In Green Computing Conference and Workshops (IGCC), 2011 International (pp. 1-9). IEEE.

[9] "Bsoul, M., Al-Khasawneh, A., Abdallah, E. E., & Kilani, Y. (2011). Enhanced fast spread replication strategy for data grid. Journal of Network and Computer Applications, 34(2), 575- 580.

[10] Cheng, Z., Luan, Z., Meng, Y., Xu, Y., Qian, D., Roy, A., & Guan, G. (2012, September). Erms: An elastic replication management system for hdfs. In Cluster Computing Workshops (CLUSTER WORKSHOPS), 2012 IEEE International Conference on (pp. 32-40). IEEE.

[11] Kousiouris, G., Vafiadis, G., & Varvarigou, T. (2013, October). Enabling proactive data management in virtualized hadoop clusters based on predicted data activity patterns. In P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2013 Eighth International Conference on (pp. 1-8). IEEE.

[12] Papoulis, A. (1977). Signal analysis (Vol. 191). New York: McGraw-Hill.

[13] Bui, D. M., Hussain, S., Huh, E. N., & Lee, S. (2016). Adaptive Replication Management in HDFS

based on Supervised Learning. IEEE Transactions on Knowledge and Data Engineering, 28(6), 1369-1382.

[14] Qu, K., Meng, L., & Yang, Y. (2016, August). A dynamic replica strategy based on Markov model for hadoop distributed file system (HDFS). In Cloud Computing and Intelligence Systems (CCIS), 2016 4th International Conference on (pp. 337-342). IEEE.

[15] Reed, Irving S.; Solomon, Gustave (1960), Polynomial Codes over Certain Finite Fields, Journal of the Society for Industrial and Applied Mathematics(SIAM), 8doi:10.1137/0108018

AUTHOR *DETAILS*

ALAPATI JANARDHANA RAO is working an assistant professor in VIGNAN'S LARA INSTITUTE OF TECHNOLOGY & SCIENCE. Vadlamudi-522213 Guntur Dist. he has Experience in the teaching field For 7 years and he interested in research areas are network security , machine learning and artificial intelligence, big data.

KOPPOLU SREE VENKATA GOPI NAGA RAJU is PURSUING MCA. In VIGNAN'S LARA INSTITUTE OF TECHNOLOGY& SCIENCE, Vadlamudi, Guntur Dist Andhra Pradesh, India