Cloud²HDD: Large-Scale HDD Data Analysis on Cloud for Cloud Datacenters

Engin Zeydan

Centre Technologic de Telecomunicacions de Catalunya,
Castelldefels, Barcelona, Spain, 08860.

E-mail: engin.zeydan@cttc.cat

Suayb S. Arslan

Department of Computer Engineering, MEF University,
Istanbul, Turkey 34912.

E-mail: arslans@mef.edu.tr

Abstract—The main focus of this paper is to develop a distributed large scale data analysis platform for the open-source data of Backblaze cloud datacenter which consists of operational hard disk drive (HDD) information collected over an observable period of 2272 days (over 74 months). To carefully analyze the intrinsic characteristics of the hard disk behavior, we have exploited a large bolume of data and the benefits of Hadoop ecosystem as our big data processing engine. In other words, we have utilized a special distributed scheme on cloud for cloud HDD data, which is termed as Cloud²HDD. To classify the remaining lifetime of hard disk drives based on health indicators such as in-built S.M.A.R.T (Self-Monitoring, Analysis, and Reporting Technology) features, we used some of the state-of-the-art classification algorithms and compared their accuracy, precision, and recall rates simultaneously. In addition, importance of various S.M.A.R.T. features in predicting the true remaining lifetime of HDDs are identified. For instance, our analysis results indicate that Random Forest Classifier (RFC) can yield up to 94% accuracy with the highest precision and recall at a reasonable time by classifying the remaining lifetime of drives into one of three different classes, namely *critical*, *high* and *low ideal* states in comparison to other classification approaches based on a specific subset of S.M.A.R.T. features.

Index Terms—Cloud, Data Center, Hadoop, HDDs, lifetime, machine learning.

I. INTRODUCTION

Hard Disk Drive (HDD) technology is still the core/most common data storage units of today's data centers which underlies the cloud storage infrastructure. Drives that operate in close proximity and share the same hardware components and backend are severely influenced by similar environmental factors and manufacturing defects which increases the likelihood of these devices experiencing similar issues or correlated damage scenarios [1]. Many techniques have been proposed in lietrature to improve cloud storage systems against such failures such as information fragmentation [2], and reinforcement learning [3]. Recently, data modeling attempts are made in order to incorporate more drive-related parameters and failure data in the durability prediction of arrays of hard drives [4], [5]. On the other hand, the methods and technology used by manufacturers during the building process can create a faulty connection between distinct storage devices. Therefore, a hardware or network problem can cause multiple storage

Fig. 1: Daily time series plot of number of measurements versus years in Backblaze's data center.

devices to fail or become unavailable simultaneously in the same network.

In general, since the reliability function (namely $R(t)$) is closely related to the cumulative distribution function (CDF) (call it $F(t)$ where $R(t) = 1 - F(t)$), it is essential to estimate the Cumulative Distribution Function (CDF) of failures in order to quantify the reliability of constituent storage devices. In particular, it is of crucial interest to accurately determine the remaining lifetime of constituent disk drives of the data center for preventive action. There are a number of challenges facing a reliability analyst to overcome for the prediction of the remaining lifetime of the constituent devices of the data center. One of most important is the hardness of the prediction model which has to take into account the correlated error scenarios as well as the usage and workload pattern. This problem can be relieved by studying the accumulated data over a time period in which the correlation of failures would be captured by the collected data itself [5]. However as the volume of data exceeds the available resources of a single machine, it became mandatory to analyze data on the cloud (computing) for the cloud (storage) systems such as big data centers [6]. For instance Fig. 1 shows the analysis of timely variations of number of measurements in a cloud data center of Backblaze¹. In general, we observe that there is

an increasing trend of collecting more measurements (except on dates 2017/01/31, 2017/01/28 and 2017/01/29 which have number of observations below 50) and its volume is exponentially increasing.

Furthermore, accurate prediction of drive lifetime from real-time health monitoring tools such as Self-Monitoring, Analysis and Reporting Technology (SMART) (now built-in to these devices and shipped with no additional cost) and comparative studies with performance metrics that are usually advertised on web would provide useful information about the product quality and reliability estimation of different brands. This comparison would also allow data system managers to evaluate the consistency and make right decisions for future business directions. Through intelligent crawling techniques on web data, the results of this paper on the drive lifetime predictions can be correlated to draw useful conclusions on prospective purchase decisions and save companies from allocating excessive budget on unnecessary and unintended storage components.

Disk failure and health monitoring data were previously collected by distinct sources and analyzed with various machine learning techniques [7], [8]. On the other hand, subsequent research works are observed to be either data size or number of models constrained [9] or else they are publicly unavailable [10]. More recently, in a number of studies, Backblaze's data is picked up and analyzed locally [11], [12]. In [11], a predictive model is proposed based on 17-month long collected data and 30000 disk data though the dataset has grown tremendously since then to encompass much richer SMART data and more disk failures. Also, regularized greedy forests is used as the classification algorithm rather than linear counterparts. It is indicated in the same study that focusing on small set of SMART indicators may result in the number of disk failures one could correctly identify drop by almost 50%. On the other hand in [12], the data set evolved to contain 47,000 drives, exhibiting hard drive heterogeneity with 81 models from 5 manufacturers. The most notable contribution of this paper was its focus on class imbalance problem which can be remedied by sampling and considering more SMART parameters. Similarly, a subset of SMART parameters are considered using classifiers based on Support Vector Machine (SVM) with linear kernel, Random Forest Classifier (RFC) and Gradient-Boosted Tree (GBT) classifier. Common to those studies was that standard classification (i.e., SVM, RFC, GBT classifier) results are obtained on a limited subset of filtered data (until 2017, recent data with more SMART values were not present). In addition none of these studies considered neural networks to efficiently utilize hidden information within some of the SMART parameters which are considered irrelevant in a proactive system design. Considering the data set is evolving every year, an adaptive and cloud-based solution is strictly needed which analyzes the collected data comprehensively with smart sampling techniques. In one of our previous studies, we have already considered failure data collected within a limited time frame to develop a visualization platform [13] merely because later data was not available at the time. However, no health monitoring output such as SMART data was utilized in the

analysis which made the predictive model less of a use from the available data point of view.

There are many tools available for large-scale data processing for batch, stream and interactive data analytic purposes [14]. For batch mode; Hadoop, HBase and Hive, Elasticsearch and for speed/stream mode; Apache Spark, Flink, Kafka, Pulsar and Redis are some of the prominent and ubiquitously used tools in the industry. In this study, we focus on Hadoop-based ecosystem due to their versatile subcomponents (Apache Hive, HDFS, Apache Spark, etc), being open source and flexible integration capabilities with visualization components. The contributions of the paper are as follows. First, we use utilize cloud computing tools for improving cloud storage technology by means of various Machine Learning (ML) techniques applied to 2272 days of collected hard drive data. To our knowledge, this is the most capacious disk drive data ever used and analyzed so far. In addition, a complete architecture containing cloud services, data analysis as well as data storage layer is presented that offers a unified and accurate analysis on very large collected data blocks. Finally, augmented SMART data (by judiciously filling missing SMART entries) with various ML algorithms are combined to classify the remaining lifetime of hard disk drive for potential preventive actions, accuracies reaching up to at least 94%.

The main contributions of the paper can be summarized as follows: (i) Utilization of a distributed large scale analysis platform to analyze collected Hard Disk Drive (HDD) information of Backblaze cloud data center over a period of 2272 days. (ii) Analysis of the intrinsic characteristics of cloud HDD behaviour based on the remaining hard disk lifetime using distinct SMART attributes as the input features. (iii) Comparison of various ML classification algorithms in terms of their classification accuracy, precision, recall rates and the time using three classification states, i.e. critical, high ideal and low ideal for different HDD vendors.

II. DATASET AND ARCHITECTURE

A. Dataset

Backblaze gathers daily measurements of drive health of each operational hard disk at their data center with quarterly updates. The fields of the daily reports include the date of the report, the serial number of the drive, the model of the drive, the capacity of the drive in bytes, a failure label that is at 0 as long as the drive is healthy and that is set to 1 when the drive fails and finally various other SMART parameters. These SMART parameters may include counts of read errors, write faults, the temperature of the drive and its reallocated sectors count, etc.². There are set of challenges when we would like to use SMART data in our predictive model. First of all, due to implementation freedom, the SMART data available for some brand may not be available for another brands resulting in a serious of blank fields. Moreover due to different implementations, the same SMART data would mean different performance indications about the health of the drive.

²<https://en.wikipedia.org/wiki/S.M.A.R.T>.

Fig. 2: General architecture and corresponding steps for data processing and analysis pipeline of Backblaze's cloud hard drive data statistics.

B. Data Analysis Architecture

Let us describe the general architecture of cloud data storage analytics platform. As shown in Fig. 2, cloud data storage analytics platform is interacting with cloud infrastructure via mainly six steps. In the first step, the cloud data analysis platform acquires the required data. This layer is called connect and transfer layer. In this layer, the main focus is on sources of data. Some of the engineering challenges that need to be solved during this layer are determining the format of data (which could be raw bytes, text files or databases), finding out the best methods for acquiring data (e.g. via interfaces either using standard protocols JDBC, HTTP, REST or customized protocols depending on the origin) as well as determining the modes of data acquisition (either batch or real-time mode). During this acquisition stage, data source is connected and raw data is transported to the next layer. In this paper, during this layer processing we connect to Backblaze website and transfer all historical data into data preprocessing layer in step-1.

Data preprocessing layer in step-2 pre-process and transforms the data so that it can be suitable for storage in different entities inside data storage layer. During this stage, all outliers (e.g. measurement cases when storage capacity less than zero, etc) are eliminated and proper transformation of columns (e.g. numeric and string transformation of all objects

of the columns, replacing nan values with the mean values of each feature) are performed. Later, in step-3, pre-processed or cleansed data are stored in appropriate data store of the cloud data analysis platform. This can be either a distributed data storage framework such as Hadoop Distributed File System (HDFS), a NoSQL database such as Cassandra, or in a file system. In this paper, we used HDFS for distributed storage of Backblaze's cloud data statistics.

Later in step-4, data analytic layer would be analyzing the data stored in the storage layer. The analysis can vary depending on the requirements of the applications and services that are running as given in step-5. Some analysis examples in data analytic layer are density estimation of hard drive failure rates (e.g. mean, median, variance, Annualized failure rate (AFR), Mean Time Between Failure (MTBF), etc), statistical analysis of the remaining lifetime and developing machine learning models for time-series analysis and prediction. In this paper, we have used Apache Spark framework in data analytic layer of step-4. In step-5 layer, cloud services and applications would be running where various services such as hard disk recommendation systems, cloud capacity or configuration management for IT managers, hard disk storage vendors or service providers might be present. This layer is interacting with a data analytics platform to query batch data,

get interactive responses based on the business requirements. Finally in step-6, if an action is required on the data storage infrastructure, it is executed through interacting with the relevant responsibilities, e.g. service providers, storage vendor companies, or end-users of each storage device.

III. EXPERIMENTAL SET-UP AND DATA CHARACTERISTICS

In this section, we obtain the HDD statistics from the BackBlaze website [15]. Our analysis platforms consists of components of Hadoop ecosystem [16]. HDFS is used for big data storage and Apache Spark framework [17] is used for distributed computing purposes. After analysis with Apache Spark framework on reducing the size of the data, we have used sklearn's ML library for classification purposes [18].

Table I provides some statistics values of the large scale hard drive data. The observation period is from 04-10-2013 until 06-30-2019 covering 2272 days. The number of features inside the raw dataset is 85 which includes columns such as date, serial number, model, storage capacity, failure status and 80 SMART indicators. The failure column becomes one when failure of storage device occurs during the observation period. The lifetime of each serial number is calculated by observing the failure column and recording the number of days until it changes from zero to one. The lifetime median value is also given with its Confidence Interval (CI) range in Table I.

A. Analysis of LifeTime

The lifetime span of a hard disk is the time span until which failure occurs for a given hard disk with serial number. In this subsection, we draw lifetime distributions CDF and histogram plots as well as by observing the change in remaining lifetime of each hard disk grouped by their serial number, we plot remaining lifetime histogram distribution. The remaining lifetime is calculated based on the lifetime distribution and represents how many days are remaining until a failure occurs for each serial number. Fig. 3a shows the CDF plot of lifetime distribution for different brands. From Fig. 3a, we can observe that Hitachi brands have higher lifetime whereas Toshiba brand tends to perform poor due to low lifetime distributions. For example, the percentage of hard disks with lifetime higher than 500 days is around 70% for Hitachi brands whereas it is around 40% for Toshiba brand. Notice also that some of the lifetime observations are small, e.g. only single observation (which is around 250 days) in case of Samsung brand. Fig. 3b shows the histogram plot of lifetime distribution as well as the fitted Kernel Density Estimation (KDE) plot. This figure shows that the most frequent lifetime is on the order of 500 days and the KDE distribution can fit to Gaussian distribution. On the other hand, Fig. 3c shows the histogram plot of remaining lifetime distribution as well as its corresponding fitted KDE plot. The most frequent remaining lifetime value is zero which corresponds to number of failures in the dataset. Moreover, remaining lifetime distribution exhibits a linear decreasing trend as can also be observed from KDE fit plot.

B. Classification of Remaining LifeTime

In this section, we first describe the results of applying ML classification algorithms to classify the remaining lifetime of the hard disk using SMART attributes as the features for the ML algorithms. The detailed explanations and meanings of each SMART features can be found in [19]. For our analysis, we defined three states of hard drives based on remaining lifetime value (where the histogram of its distribution is already given in Fig. 3c). Those states are defined as critical, low and high ideal states. Mainly, if remaining lifetime is between $[0, 219]$ days it is in critical state, if it is between $(219, 519]$ days it is in low ideal state and if it is between $(519, 1962]$ it is in high ideal state. These demarcation values on remaining lifetime are selected based on the optimization criteria that keeps number of measurements observed in each of these three states the same (around 1,800,000 measurements in each state). We have used 80% of data for training and 20% for testing purposes. Note that not all of the SMART features are available for all models. For example, Seagate and Hitachi brands have all NaN values on 2 and 34 of the SMART features respectively. Hence, for application of ML algorithms, we have grouped the dataset so that separate models can be trained for each of brands. Without loss of generality, our focus has been on Seagate and Hitachi which have the most number of measurements in the database (around 79% of all measurements). We have used Logistic Regression (LR), Decision Tree (DT), Naive Bayes (NB), GBT and RFCs. The main reasons of selecting these ML algorithms are their wide range of applications on real world problems, easier tuning properties with minimum hyper-parameter optimization and reasonably high predictive accuracy power in high dimensional data. For our ML algorithms, we have used all available 81 of the numerical features which includes 80 SMART indicators and one storage capacity feature to classify the target class of remaining lifetime, i.e. high-ideal, low-ideal and critical state of a given measurement.

For visualization purposes, Fig. 4 shows the t-SNE (t-Distributed Stochastic Neighbor Embedding) plot of 20,000 randomly selected measurements (out of 289,764) with 85 features and their corresponding classification states, i.e. critical, high ideal and low ideal that are reduced to two dimensions for Hitachi brand. t-SNE is a nonlinear dimensionality reduction technique that is used to visualize high dimensional data in a low dimensional (either two or three) space. It is mainly used to model similar observation data points via nearby point locations whereas dissimilar ones via distant point locations with high probability [20]. Based on these facts, Fig. 4 clearly indicates that ML algorithms need to classify the visualized complex interrelations between 81 different features that are available in large scale measurements.

Logistic Regression Classification: After grid-search optimization, the best hyper-parameters for LR are selected to be $max_iter = 50$ (the maximum number of passes over the training data (aka epochs)) and the accuracy level is around 0.53 and 0.55 for both Hitachi and Seagate brand's

TABLE I: Some statistics information about the analyzed large-scale hard drive dataset.

Total # of measurements	146,235,260	# of features	85
Total # of failures	9,734	LifeTime (median CI (95%), days)	[488.62, 505.38]
Total # of different brands	6	Storage capacity (median, TB)	12
Total # of unique serial numbers	167,888	# of unique storage capacity size	20
Obser. period (days)	2272	Total volume of data (raw / after preprocessing)	34.64 GB / 1.6 GB
Storage capacity (failed drives) (median, TB)	4	# of days (after pre-processing)	1961
# of features (using for classification)	81	# of measurements (after pre-processing)	146,230,032

Fig. 3: Lifetime distribution for different brands (a) CDF plot. (b) Histogram and KDE plots (c) Remaining Lifetime histogram and KDE plot.

TABLE II

ACCURACY, PRECISION, RECALL, F1 SCORE, AUC AND CLASSIFICATION TIME OF DIFFERENT CLASSIFIERS FOR SEAGATE BRAND.

Classifier	Accuracy	Precision	Recall	F1 Score	AUC	Classification Time
LR	0.55	0.55	0.55	0.55	0.66	5 min 21 sec.
NB	0.51	0.54	0.52	0.51	0.64	17 sec.
DT	0.81	0.81	0.81	0.81	0.86	4 min 23 sec.
GBT	0.77	0.77	0.77	0.77	0.82	5 hours 48 min
RFC	0.94	0.94	0.94	0.94	0.95	2 min 48 sec.

TABLE III

ACCURACY, PRECISION, RECALL, F1 SCORE, AUC AND CLASSIFICATION TIME OF DIFFERENT CLASSIFIERS FOR HITACHI BRAND.

Classifier	Accuracy	Precision	Recall	F1 Score	AUC	Classification Time
LR	0.53	0.53	0.53	0.53	0.65	8.7 sec.
NB	0.51	0.51	0.51	0.51	0.63	1.33 sec
DT	0.91	0.92	0.91	0.91	0.93	3.68 sec.
GBT	0.91	0.92	0.91	0.91	0.94	253.28 sec.
RFC	0.92	0.92	0.92	0.92	0.94	30 sec.

measurements respectively.

Naive Bayes Classification: We use NB classifier for multivariate Bernoulli models for $\alpha = 1.0$ (additive (Laplace/Lidstone) smoothing parameter). The accuracy scores are relatively low but the classification time is fast. For both Hitachi and Seagate, the accuracy scores are around 0.51 with classification time of 1.33 sec. and 17 sec. respectively.

Decision Tree Classifier: For both Hitachi and Seagate, the hyper-parameters are selected to be $max_depth = 23$ (the maximum depth of the tree) and $min_samples_leaf = 1$ (the minimum number of samples required to be at a leaf node). For Hitachi, the accuracy is around 91%. Top 3 features that are important for classification are *smart_9_raw*

(Power-On Hours), *smart_3_normalized* (Spin-Up Time) and *smart_5_raw* (Reallocated Sectors Count) with importance ratios of 0.19, 0.10 and 0.08 respectively. For Seagate, the accuracy is comparatively low and is around 81%. Top 3 features that are important for classification are *smart_9_raw* (Power-On Hours), *smart_194_raw* (Temperature) and *smart_200_normalized* (Multi-Zone Error Rate) with importance ratios of 0.13, 0.10 and 0.07 respectively.

Gradient Boosting Tree Classifier: The best hyper-parameters for GBT classifier are selected to be $n_estimators = 100$ (the number of boosting stages to perform), $max_depth = 7$ (maximum depth of the individual regression estimators), $learning_rate = 1$ (the

Fig. 4: t-SNE plot for Hitachi brand measurements with corresponding three states having dimensions reduced to two.

rate shrinks the contribution of each tree) for Hitachi brand measurements. The accuracy of GBT classifier is observed to be around 92%. The top 3 features of importance are *smart_9_raw* (Power-On Hours), *smart_3_raw* (Spin-Up Time) and *smart_5_raw* (Reallocated Sectors Count) with importance ratios of 0.15, 0.08 and 0.07 respectively. The hyper-parameters of GBT for Seagate are selected to be $n_estimators = 50$, $max_depth = 7$, $learning_rate = 1$. The accuracy of GBT classifier is observed to be around 72%. The top 3 features of importance are *smart_9_raw* (Power-On Hours), *smart_188_raw* (Command Timeout) and *smart_225_raw* (Load/Unload Cycle Count) with importance ratios of 0.20, 0.20 and 0.06 respectively.

Random Forest Classifier: After grid-search, the best hyper-parameters for RFC are selected to be $n_estimators = 100$ (the number of trees in the forest), $maxdepth = 30$ (the maximum depth of the tree) for Hitachi brand and $n_estimators = 100$, $maxdepth = 50$ for Seagate brand dataset. The accuracy of RFC is observed to be the highest with 92% in Hitachi and 94% for Seagate brand considering the performance of all other considered ML algorithms. Fig. 5 shows top 16 most important features of RFC during classification process. The top important features for Hitachi brand are *smart_9_raw* (which corresponds to Power-On Hours), *smart_5_raw* (which corresponds to Reallocated Sectors Count) and *smart_194_raw* (which corresponds to Temperature) with corresponding importance ratios of 0.2, 0.07 and 0.07 respectively. The top important features for Seagate brand are *smart_9_raw* (which corresponds to Power-On Hours), *smart_194_raw* (which corresponds to Temperature) and *smart_1_raw* (which corresponds to Read Error Rate) with corresponding importance ratios of 0.12, 0.08 and 0.06 respectively.

In summary, Table II and Table III show the accuracy, precision, recall, F1 score, AUC and classification time of different classifiers over the test dataset for Seagate and Hitachi brands respectively. Note that the classification time is obtained via sklearn library on a single machine. AUC (area under the ROC curve) scores are evaluated for multi-class problem by

Fig. 5: Random forest feature importance of top 16 features.

binarizing the labels in one versus all manner and averaging the results of prediction scores of binary classification tasks. To get accuracy, precision, recall and F1 score metrics, we used each label's unweighted mean.

C. Discussions on Analysis Results

The above described results indicate that RFC yields the highest accuracy in less classification time for both Seagate and Hitachi brands. On the other hand, GBT classifier yields similar accuracy results for Seagate but with higher classification time. Therefore, RFC performance is more reasonable considering the accuracy and classification accuracy trade-offs. In addition RFC provides sizable better recall and precision rates indicating both measures are treated equally and well. The same classifier can be configured to weight one of them over the other in the training phase depending on the use cases of the system. We also note that in all the above top three feature importance values of DT, GBT and RFC, the storage capacity does not represent the highest differentiating factor (e.g. as given in Fig. 5).

As it is clear from Tables I and II, detection of hard drive failures is of key importance to take preventive action up ahead if there is really a failure. In case of failure to detect it, due to other failure detection mechanisms embedded in cloud systems, we will only delay the overall repair process. On the other hand, detecting a failure for a healthy drive will initiate a needless recovery process which shall introduce an extra repair burden to the system. The latter would lead to an increase in the complexity and communication cost. Depending on the availability of other failure detection mechanisms or the time it takes to repair a drive, recall (due to potential data corruption) or precision (due to extra complexity burden) rate may be more important. Hence, it is a question of preference on latency or complexity to trade recall and precision rates one over the other.

IV. CONCLUSIONS AND FUTURE WORK

In this paper, we have analyzed large-scale hard disk drive dataset accumulated from different hard drive vendors using Hadoop-based big data processing platform over an observation period of 2272 days. In other words, we have utilized

a special distributed storage and computing scheme on *cloud for cloud* which shall prove useful for offloading both storage and computation to the cloud for efficient analytics workloads. For the analysis part, we primarily described the general architecture of the cloud data storage analytics platform that consists of open source distributed large-scale data storage, processing and analysis frameworks using components from Hadoop ecosystem namely HDFS and Apache Spark. Then, based on the remaining lifetime calculations, we categorized the state of each measurement into three different states namely high, low ideal and critical. Finally, we compared the classification performance of various classification algorithms based on the remaining lifetime using SMART features of the HDD data in Backblaze's cloud data center infrastructure. Our analysis results indicated that up to 94% and 92% accuracy values can be achieved with RFC algorithm in reasonable classification time compared to other considered ML classification algorithms for Seagate and Hitachi brands, respectively. Our proposed large-scale cloud data processing framework together with the proposed system architecture and initial analysis results can be used as a guideline for data center engineers, IT practitioners and reliability engineers that might be confronted with regular disk storage failures everyday.

As a future work, we are planning on extending our analysis by applying modern deep learning techniques to our classification/prediction problem. Since the performance of artificial neural networks (ANNs) are heavily dependent on the size and feature set quality, we shall study the best describing SMART features for potential preventive action by gathering more field data. Furthermore as we have noticed the training time may be increasing as the size of the data set grows, we shall be distributing the training computation workload over multiple workers to speed up the training process.

ACKNOWLEDGMENT

This work was partially funded by The Scientific and Technological Research Council of Turkey (TUBITAK) under the grant number 2232-115C111, Spanish MINECO under the grant number TEC2017-88373-R (5G-REFINE) and by Generalitat de Catalunya under the grant number 2017SGR1195.

REFERENCES

[1] B. Schroeder and G. A. Gibson, "Disk failures in the real world: What does an mttf of 1 000 000 hours mean to you?," in *Proc. 5th USENIX Conf. FAST*, pp. 1–16, 2007.

[2] H. Qiu, G. Memmi, and H. Noura, "An efficient secure storage scheme based on information fragmentation," in *2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)*, pp. 108–113, IEEE, 2017.

[3] R. R. Noel, R. Mehra, and P. Lama, "Towards self-managing cloud storage with reinforcement learning," in *2019 IEEE International Conference on Cloud Engineering (IC2E)*, pp. 34–44, IEEE, 2019.

[4] S. S. Arslan, "A reliability model for dependent and distributed mds disk array units," *IEEE Transactions on Reliability*, vol. 68, pp. 133–148, March 2019.

[5] S. S. Arslan, J. Peng, and T. Goker, "A data-assisted reliability model for carrier-assisted cold data storage systems," *Reliability Engineering & System Safety*, vol. 196, p. 106708, 2020.

[6] H. Qiu, H. Noura, M. Qiu, Z. Ming, and G. Memmi, "A user-centric data protection method for cloud storage based on invertible dwt," *IEEE Transactions on Cloud Computing*, 2019.

[7] Y. Zhao, X. Liu, S. Gan, and W. Zheng, "Predicting disk failures with hmm-and hsmm-based approaches," in *Industrial Conference on Data Mining*, pp. 390–404, Springer, 2010.

[8] Y. Wang, Q. Miao, and M. Pecht, "Health monitoring of hard disk drive based on mahalanobis distance," in *2011 Prognostics and System Health Management Conference*, pp. 1–8, IEEE, 2011.

[9] A. Ma, R. Traylor, F. Douglass, M. Chamness, G. Lu, D. Sawyer, S. Chandra, and W. Hsu, "Raidshield: characterizing, monitoring, and proactively protecting against disk failures," *ACM Transactions on Storage (TOS)*, vol. 11, no. 4, p. 17, 2015.

[10] J. Li, X. Ji, Y. Jia, B. Zhu, G. Wang, Z. Li, and X. Liu, "Hard drive failure prediction using classification and regression trees," in *2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, pp. 383–394, IEEE, 2014.

[11] M. M. Botezatu, I. Giurgiu, J. Bogojeska, and D. Wiesmann, "Predicting disk replacement towards reliable data centers," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 39–48, ACM, 2016.

[12] N. Aussel, S. Jaulin, G. Gandon, Y. Petetin, E. Fazli, and S. Chabridon, "Predictive models of hard drive failures based on operational data," in *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 619–625, IEEE, 2017.

[13] I. O. Yigit, S. S. Arslan, and E. Zeydan, "A visualization platform for disk failure analysis," in *2018 26th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, May 2018.

[14] Y. Arfat, S. Usman, R. Mehmood, and I. Katib, "Big data tools, technologies, and applications: A survey," in *Smart Infrastructure and Applications*, pp. 453–490, Springer, 2020.

[15] Backblaze, "Hard drive data and stats." <https://www.backblaze.com/b2/hard-drive-test-data.html>, 2018. [Online; accessed 31-October-2019].

[16] Apache Hadoop, "Apache™ Hadoop® project." <http://hadoop.apache.org/>, 2019. [Online; accessed 18-October-2019].

[17] Apache Spark, "Apache Spark™ - Unified Analytics Engine for Big Data." <https://spark.apache.org/>, 2019. [Online; accessed 15-November-2019].

[18] Scikit-learn, "Scikit-learn: Machine learning in Python." <https://scikit-learn.org/stable/>, 2019. [Online; accessed 15-November-2019].

[19] S.M.A.R.T., "Self-Monitoring, Analysis and Reporting Technology." <https://en.wikipedia.org/wiki/S.M.A.R.T.>, 20198. [Online; accessed 15-November-2019].

[20] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.