Received October 24, 2018, accepted November 19, 2018, date of publication November 29, 2018,
date of current version December 27, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2884004

New Metrics for Disk Failure Prediction That Go Beyond Prediction Accuracy

JING LI[1], REBECCA J. STONES[2], GANG WANG[2], (Member, IEEE), ZHONGWEI LI[2], XIAOGUANG LIU[2], (Member, IEEE), AND JIANLI DING[1]

[1]College of Computer Science and Technology, Civil Aviation University of China, Tianjin 300300, China
[2]Nankai-Baidu Joint Laboratory, College of Computer Science, Nankai University, Tianjin 300071, China

Corresponding authors: Jing Li (lijing@nbjl.nankai.edu.cn) and Rebecca J. Stones (becky@nbjl.nankai.edu.cn)

This work was supported in part by the NSF of China under Grant 61702521, Grant 61602266, Grant 61872201, and Grant U1833114, in part by the NSF of Tianjin under Grant 4117JCYBJC15300, in part by the Science and Technology Development Plan of Tianjin under Grant 17JCYBJC15300 and Grant 16JCYBJC41900, in part by the Scientific Research Foundation of Civil Aviation University of China under Grant 2017QD03S, in part by the Fundamental Research Funds for the Central Universities, and in part by the SAFEA: Overseas Young Talents in Cultural and Educational Sector. The work of R. J. Stones was supported by the Thousand Youth Talents Plan in Tianjin.

ABSTRACT Prediction accuracy (true positives, false positives, and so on) is the usual way for evaluating disk-failure prediction models. Realistically however, we aim not only to correctly predict failures, but also to protect data against failure, i.e., we need to take appropriate action after a failure prediction. In the context of storage systems, protecting data requires that we migrate at-risk data, but this consumes network and disk bandwidth, which is particularly problematic for large-scale and cloud systems. This paper consolidates and builds on Li *et al.* (2016), where we propose using two new metrics, migration rate (MR) and mismigration rate (MMR), to measure the quality of disk failure prediction: MR measures how much at-risk data is migrated (and therefore protected) as a result of correct failure predictions, while MMR measures how much data is migrated needlessly as a result of incorrect failure predictions. In this paper, we additionally propose measuring quality in terms of migration time and mismigration time, which measure the time spent migrating at-risk disks, and the time spent mismigrating healthy disks caused by false alarms, respectively. To demonstrate these metrics' usefulness, we use them to compare disk-failure prediction methods: we compare: 1) a classification tree (CT) model against a state-of-the-art recurrent neural network (RNN) model and 2) a gradient-boosted regression tree (GBRT) model (which predicts residual life) against RNN. We observe that while RNN performs best in the prediction accuracy experiments, the CT and GBRT models sometimes outperform RNN in the resource-dependent migration-rate experiments. We conclude that prediction accuracy is sometimes misleading: correct predictions do not necessarily imply protected data. We additionally present an improved GBRT model (GBRT+), which offers a practical improvement in disk residual-life prediction accordingly to the newly proposed metrics.

INDEX TERMS Disk failure prediction, evaluation metrics, migration accuracy, resource consumption, cloud storage system.

I. INTRODUCTION

Data centers are arguably the most important infrastructure in the era of cloud computing, with hard disks ordinarily being their primary data storage devices [1]–[3]. However, hard disks are subject to disk failure, which leads to reduced service availability (e.g., downtime), and possibly even in permanent loss of data, which hurt the user experience. Therefore, reliability is by far the most serious concern in current data centers. In addition to reactive fault tolerance techniques (typically erasure codes and data replication), proactive fault tolerance is used to improve a storage system's reliability: instead of waiting for failures to occur and then responding, with proactive fault tolerance we predict failures in advance, and thereby enable the system or operator to respond appropriately.

Prior work proposed various statistical and machine-learning methods for building disk-failure prediction models which utilize SMART (Self-Monitoring, Analysis and Reporting Technology) attributes [4]–[17]. To summarize the situation:

- Previous research [4]–[15] focused on predicting whether or not a disk will fail in the near future.

Failure detection rate (FDR) and *false alarm rate* (FAR) are used to measure a model's *classification accuracy* (as binary classifiers); FDR measures the proportion of at-risk disks that are predicted to fail in the near future, and FAR measures the proportion of failure predictions which are false (i.e., the disks are healthy).

- Later prediction models [16], [17] predict the remaining working time (i.e., residual life) of a disk, so that system administrators can allocate system resources more effectively in response to a disk failure pre-warning, ensuring the reliability and availability of system. An *accuracy of residual-life level assessment* (ACC) is used to measure a model's *classification accuracy* (as multiple classifiers); the remaining working time predictions are partitioned into levels (or intervals), and the ACC is defined as the proportion of predictions which fall into the correct interval.

The existing metrics mainly measure a model's classification accuracy using general prognostic techniques, isolated from their application. In the context of large-scale and cloud storage systems, disk failures and their corresponding failure-warnings occur sufficiently frequently so that pre-warning migration is a continuous resource drain. All else being equal, a higher classification accuracy is always beneficial. However, in practice improving classification accuracy involves various trade-offs, such as the *time in advance* (TIA), i.e., the mean warning time. For example, prediction accuracy may be artificially inflated to nearly 100% at the expense of TIA, e.g. by reducing it to one hour. In this case, the at-risk data is incapable of being completely protected before failure, even if the failure is predicted in advance. Thus, classification accuracy alone does not give the whole picture: there are other relevant factors involved.

Moreover, when an alarm is raised, the system migrates at-risk data to other healthy disks through the network (even crossing top-of-rack switches, if disks are stored in different racks), which increases the network load, thereby decreasing the bandwidth available for user services. With more bandwidth available for migration (or higher migration transfer rates), we protect more at-risk data before actual failure occurs, but we incur worse effects on the quality of user services. In other words, migration resource consumption determines not only the amount of data protected by prediction models, but also the impact of migrations on the quality of user services. Therefore, it is useful to also take resource consumption into consideration when evaluating disk-failure prediction models.

This paper expands on previous work involving most of the present authors' [18]. In [18], (a) we propose measuring the data protection of disk failure prediction models in terms of *migration rate* (MR) and *mismigration rate* (MMR), defined as the proportion of data on at-risk disks that are successfully migrated, and the proportion of data on healthy disks which are needlessly migrated, respectively, and (b) we propose a residual-life prediction model based on gradient-boosted regression trees (GBRTs). We make the following additional contributions:

- Motivated by how data protection depends on resource consumption, we further propose measuring the resource consumption of migrating data due to disk failure predictions, in terms of *migration time* (MT) and *mismigration time* (MMT), defined as the mean time to migrate data from an at-risk disk, and the mean time to migrate data from a disk falsely predicted to fail. These metrics reflect the mean migration transfer rate and the impact of migrations on user services. Data-protection performance, however it is measured, varies as the migration rate varies. This property is accounted for in migration accuracy (MR and MMR) and in migration time (MT and MMT).
- With the aim of protecting more at-risk data (i.e., higher migration accuracy) with less resource consumption (i.e., longer migration time), we modify the training algorithm for the GBRT model to focus on accurately protecting data rather than only predicting the residual life for each disk (Section III-C.3). Moreover, based on experiments on a real-world dataset, we observe that the modified GBRT model (GBRT+) outperforms the original GBRT in terms of migration accuracy and resource consumption (Table 8).
- We enhance the exposition and give explicit formulas for MR and MMR; see (2) and (3).
- We discuss the trade-off between data protection and resource consumption for disk failure prediction models (Section V-A.3).

For general prognostic models, Saxena *et al.* [19], [20] (see also [21]) surveyed many metrics for measuring accuracy in various ways for various applications; they assigned them to three categories: accuracy, precision, and cost/benefit. In this paper, we focus on cost/benefit metrics: the benefit is avoiding data loss at the cost of unnecessary resource utilization, which is similar to the "technical value" metric in [22].

The remainder of the paper is organized as follows: Section II surveys related work on SMART-based disk-failure prediction and their evaluation metrics. In Section III, we introduce the four metrics (MR, MMR, MT, and MMT) and the relevant disk residual-life prediction models. In Section IV we describe the datasets and their curation. Section V gives the experimental results, and Section VI concludes the paper.

II. RELATED WORK

Proactive fault tolerance predicts impending disk failures enabling the system (or an operator) to take actions in advance to prevent data loss (ordinarily data migration), thereby enhancing the reliability and availability of storage system significantly. Prior work focused on SMART-based disk failure prediction. However, SMART threshold-based inbuilt failure prediction is capable of only achieving an FDR of around 3% to 10% with a FAR of around 0.1% [4], assuming a practicable (and hence conservative) FAR.

Prediction performance has been enhanced in various ways, such as adopting Bayesian approaches [4], [5], hidden Markov models [8], applying the Wilcoxon rank-sum test [6], [7], Mahalanobis distance [10], exploring Backpropagation artificial neural networks [9], a Gaussian mixture model [23], and classification trees [12], [15], [24]. Almost all prior research treats disk failure prediction as a simple binary classification: a disk is either ''healthy'' or ''at-risk''. The quality of previously proposed methods has been measured by correctly identifying failures (in terms of FDR), while avoiding false positives (in terms of FAR) and having a practicable time in advance (TIA), which is neither too long nor too short.

In practice, disk failures tend to occur gradually (see e.g. [9], [12]) and signs of deterioration are present in some SMART attributes. In previous work involving the present authors, we explored a regression-tree-based disk health-degree prediction model [12] where a disk's ''health degree'' was defined as its failure probability. Furthermore, a combined Bayesian network model [17] was explored predicting the *residual life* of a disk, measured using *classification precision*, which was defined as the proportion of disks predicted into correct residual-life level. Motivated by the observation that the health statuses of disks have long-range dependency, Xu *et al.* [16] proposed a recurrent neural network (RNN) based method, measured using health-level classification accuracy.

All previous work treated disk failure prediction as a binary or multiple classification issue, and the metrics used in them are assessed according to whether or not their classifications are accurate (measuring true positives and false positives [21]), which isolated the prediction models from the practical application in industry: ''accuracy'' does not entail ''usefulness''. The usual way of overcoming this problem is by also measuring the warning time (time in advance, TIA) but this is a problematic metric [25]. To illustrate, if we predict every disk will fail and simply wait until they all fail, we obtain perfect accuracy with huge TIA. However, while this method is 100% ''accurate'', it is worse than useless: it incurs unnecessary migration and disk replacement costs.

Ultimately the goal predicting disk failure is to avoid or reduce data loss, which requires two steps: (1) correctly predicting which disks are about to fail; and (2) timely completing the resource-dependent pre-warning migration processes. Or more generally:

> The ultimate goal of prognostics algorithms is to reduce the occurrence of unscheduled maintenance. –Leao *et al.* [26]

When building disk-failure prediction models for practical storage systems, we benefit from using an evaluation metric which incorporates the completion status and resource consumption for disk pre-warning migration.

Some researchers have focused on pre-warning migration strategies after disk failure prediction: IDO [27] was proposed identifying impending failures and migrating at-risk data in ''hot zones'' to some substitute RAID set; Ma *et al.* [13] designed RAIDSHIELD to replace at-risk disks according to a joint failure probability; and Fatman [27] proactively migrates at-risk data in a hybrid system using both replicated ''hot'' data and erasure-coded ''cold'' data. The aforementioned work did not endeavor to reduce the impact of migrations on the user service, but simply migrated the data on disks which will potentially fail, as predicted by the models. Addressing migration costs, Ji *et al.* [28] proposed a pre-warning migration technique which manages the prioritization of pre-warning migrations, and ProCode (proactive erasure coding scheme) was proposed by Li *et al.* [29] to increase the number of replicas of data blocks' on at-risk disks.

Leao *et al.* [26, eq. (7)] modeled converting accuracy metrics to cost/benefit metrics. However, it differs from the present work in that their cost is directly financial, whereas the ''cost'' in this work is the system's resources. Dzakowic and Valentine [22, eq. (9)] described a ''technical value'' metric, which models the financial cost of prognostics; it incorporates fault isolation which we do not require (since we do not isolate at-risk disks), so ignoring those terms, the technical value is

$$P_f D\alpha - (1 - P_f)P_D\phi \tag{1}$$

where P_f is the probability of being in failure mode, D is the overall detection confidence, α is the savings when pre-detecting a fault, P_D is the false positive rate, and ϕ is the cost of false positives. The proposed metrics are similar to the terms $P_f D$ (for MR/MT) and $(1 - P_f)P_D$ (for MMR/MMT). However, the proposed metrics deviate from [22] (and [26]) in several ways:

1) the proposed metrics incorporate dynamic responses to failure predictions (i.e., with different levels of urgency), not just having ''maintenance'' as on/off;
2) the proposed metrics incorporate the possibility of incomplete migration due to failure occurring or due to changing predictions;
3) we measure ''cost'' in two ways: the quantity of migration (i.e., data amount) and the time spent migrating, which are related to the responsiveness of system (instead of financial costs); and
4) we do not assume a system with a single ''failure mode'', but rather we admit the possibility that disk-failure predictions and migrations overlap with one another (which better models cloud and other large-scale storage systems).

In this paper, we analyze two groups of new metrics for evaluating disk-failure prediction models, and build a residual-life prediction model which improves the GBRT model according to these metrics. The proposed metrics are application-specific: they are tailored to the problem of disk-failure prediction in large storage systems, where resource costs incurred by migration are not negligible nor ''once in a while'' considerations.

III. THE PROPOSED METHOD

A. PRE-WARNING HANDLING

In practice, disk failure prediction models running on storage systems monitor the working disks in real time, and output their health states periodically (such as hourly). After a disk is predicted to fail (in the near future), the at-risk data are ordinarily migrated to other spare or healthy disks. To reduce the impact on the user service, system operators limit the resources (network and disk bandwidth) used for pre-warning migrations. These mechanisms can be adapted to dynamically adjust the transfer rate for pre-warning migrations based on urgency, but this requires accurate residual-life prediction.

Suppose a disk s with m TB capacity will fail in h hours. To optimize the balance between the user service (i.e., ensuring the system's responsiveness) and pre-warning migration, we could set the transfer rate to around m/h TBs per hour. Provided nothing unexpected occurs, migration of the data on disk s would complete just before failure actually occurs. If migrating data directly from an at-risk disk increases the load on the disk thereby increasing the risk of failure, we may alternatively use a healthy data replica with a low load as migration source (in a replication system), or reconstruct the at-risk data using healthy redundant information (in an erasure code system), like in [29].

To avoid incomplete migration arising from inaccurate predictions, we adopt the method proposed in [16] to divide the possible residual life into several levels according to its urgency, and migrate data according to a predetermined transfer rate for each level, as listed in Table 1, motivated by [28]. For example, level 1 has the highest urgency, and implies the disk's residual life is less than one day. For longer (less urgent) residual life predictions, we choose slower transfer rates.

Table 1 is just a simple partition example, although it is not unreasonable in practice, as we see in the experiments in Section V. If the strategy in Table 1 is unsuitable, system operators can design a pre-warning handling strategy according to their system's configuration and the demand for system reliability.

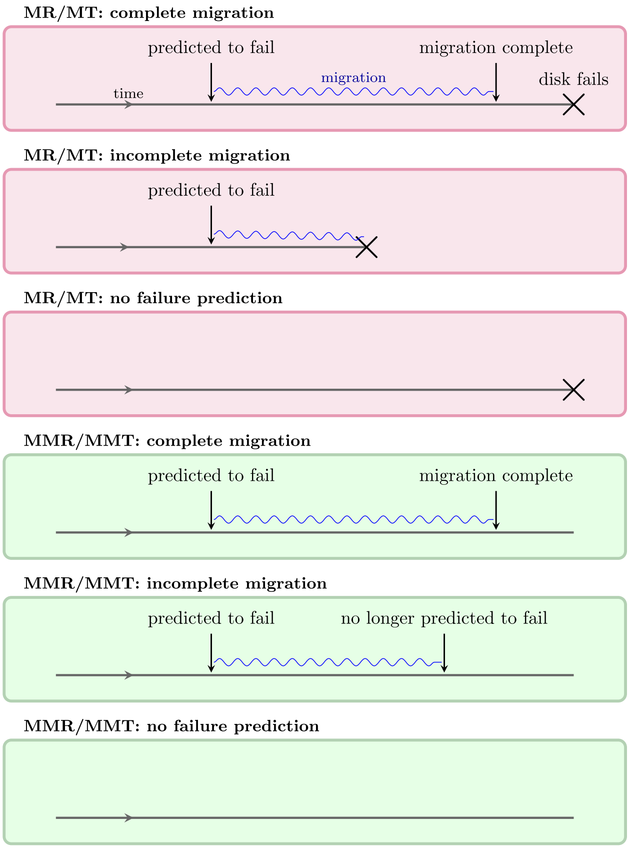
TABLE 1. Example migration transfer rates. The disk capacity is denoted *m*.

Level	Residual life (hours)	Transfer Rate (an hour)
1	$0-24$	$m/5$
2	$25-72$	$m/24$
3	$73-168$	$m/72$
4	$169-336$	$m/168$
5	$337-500$	$m/336$
6	> 500	0

In contrast, in a system using a binary classifier for disk failure prediction, all the at-risk disks are ordinarily migrated with equal urgency (i.e., their data are migrated at a uniform transfer rate). Due to the possibility of detection errors, an individual detection is insufficient to give a reliable disk-failure warning. Thus, we do not immediately raise an alarm, and instead continue to monitor the disk. Likewise, we do not immediately halt migration due to a contrary prediction.

B. NEW METRICS

When a failure is predicted, the migration for at-risk data ideally completes before failure actually occurs (which means the migration time should be shorter than the residual life of the at-risk disk), otherwise the erased data are recovered according to the reactive fault tolerance method used (or are permanently lost if recovery is not possible). We depict the possible scenarios in Figure 1: a disk may or may not encounter (a) failure prediction, (b) migration completion (after a failure prediction), and (c) actual failure.


MR/MT: complete migration

MR/MT: incomplete migration

MR/MT: no failure prediction

MMR/MMT: complete migration

MMR/MMT: incomplete migration

MMR/MMT: no failure prediction

FIGURE 1. Timelines for the six operating scenarios we consider. We measure migration either in terms of data transfers (MR and MMR) or time (MT and MMT), after being averaged over all relevant disks. The traditional metrics FDR and FAR treat "incomplete migration" as "complete migration"; they also do not incorporate the possibility of multiple separate failure predictions for a single disk.

With the advent of cloud computing and the global growth of data, current storage systems (as opposed to traditional storage systems) differ in some ways: (a) an increasingly large scale (e.g., there are hundreds of millions of disks in Azure, a typical cloud storage system [30]); (b) disk failures (and disk-failure predictions) are more frequent; and (c) disks and bandwidth resources play a major role in maintaining a quality user service. Therefore, allocating network and disk I/O for all pre-warning migrations significantly reduces system availability.

To ensure a quality user service, system operators need to limit resources used for pre-warning migrations, which increases the possibility of incomplete migration. In other words, throttling migration to conserve resources results in more incomplete migrations prior to disk failure (as in Figure 1). This is particularly pertinent when failure predictions and data migrations are the norm, as in large-scale and cloud storage systems.

To measure the data-migration resource consumption due to failure prediction (and thereby measure its effect on the quality user service), we propose two additional metrics, migration time and mismigration time, which are introduced in the introduction. Migration time (MT) and mismigration time (MMT) are a function of disk capacity and migration transfer rates (which are influenced by the available network and disk bandwidth). For example, in a cloud storage system with disks of some specific capacity, its migration time reflects the bandwidth cost for pre-warning handling.

In a system with disk residual-life prediction, the migration transfer rate may change as the residual life of the disk (predicted by model) varies; see Section III-A. So for a pre-warning handling process, there may be several transfer rates, but only a single migration time (corresponding to the mean transfer rate), which we illustrate in Figure 2.

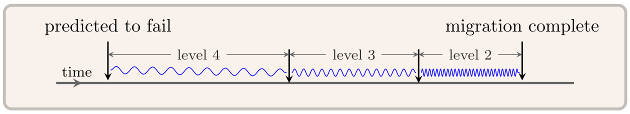


FIGURE 2. A hypothetical disk's timeline: we migrate data (blue) at a rate corresponding to the failure prediction level.

In Figure 2 we depict a toy example timeline of a disk which is predicted to fail: initially the prediction is at level 4, so we migrate data at the rate $m/168$ (as per Table 1). This is subsequently updated to level 3, so we increase the migration transfer rate to $m/72$, and again updated to level 2, so we increase the migration transfer rate to $m/24$. Dynamic migration transfer rates imply that the time spent migrating data may not correspond directly with the amount of data migrated, so we measure "migration time" separately from "migration rate".

Let F and G respectively denote the sets of all failed and all healthy disks in the system. Let F^* denote the failed-disks set for which the content is migrated to healthy disks. Let G^* denote the set of healthy disks falsely predicted to fail. To measure the time spent on pre-warning migration, we define the migration time

$$\mathrm{MT} = \frac{1}{|F^*|} \sum_{s \in F^*} m_s(P, H),$$

and the mismigration time

$$\mathrm{MMT} = \frac{1}{|G^*|} \sum_{s \in G^*} m_s(P, H),$$

where m_s denotes the time spent on migrating data from disk s based on a prediction result P and a pre-warning strategy H.

Pre-warning migration only contributes to the migration time when the transfer rate is non-zero. Thus, if the migration process is interrupted (ordinarily due to a level-6 prediction, as per Table 1), the time during the interruption is not counted towards MT or MMT. We define MT as the average time to completely migrate data from a disk. In order to incorporate partially migrated disks, which fail before migration is complete, we scale its actual process time to the expected completion time.

To measure the migration accuracy of a failure prediction model, migration rate (MR) and mismigration rate (MMR) are defined as in [18], namely

$$\mathrm{MR} = \frac{\sum_{s \in F} M_s(P, H)}{\sum_{s \in F} C_s}, \tag{2}$$

and

$$\mathrm{MMR} = \frac{\sum_{s \in G} M_s(P, H)}{\sum_{s \in G} C_s}, \tag{3}$$

where C_s is the quantity of data on disk s, and $M_s(P, H)$ is the quantity of data migrated from disk s (before it fails) based on a prediction result P and a pre-warning strategy H. Thus, the numerator of (2) (resp. (3)) is the quantity of data migrated from failed disks (resp. healthy disks) which are predicted to fail. And the denominator of (2) (resp. (3)) is the total quantity of data on failed disks (resp. healthy disks).

In practice, migration accuracy of a prediction model is affected by migration transfer rates, thus the values of MR and MMR depend on the prediction result P (which indicates a level of urgency) and pre-warning handling strategy H (which indicates what to do for a given level of urgency). This is reflected in the equations (2) and (3), but not reflected in the traditional metrics.

The two groups of measures (MR and MMR, and MT and MMT) have the following properties: (a) They are meaningful and understandable; specifically, MR/MMR measures the quantity of at-risk/healthy data that is successfully migrated (protected), and MT/MMT measures the resource consumption of migrating data. (b) They are defined proportionally, which matches how a larger system is capable of handling a greater migration load. (c) They enable comparing disk failure prediction models according to migration accuracy and resource consumption.

A higher MR generally implies more at-risk data are protected successfully, lower MMR implies less bandwidth resources are wasted on false failure predictions, and longer MT and MMT imply a smaller bandwidth cost and a lesser resource burden for pre-warning handling. Further, higher migration transfer rates (reflected as shorter MT and MMT) imply higher MR and MMR. If all the migrations complete, then MR is equal to MMR, and MMR is equal to FAR.

C. PREDICTION MODEL

We predict a disk's remaining operating life based on its SMART records; we formulate this prediction as a regression problem. In this section, we introduce a modified model based on gradient-boosted regression trees (GBRTs) [31], which outputs a quantitative target value to describe the remaining operating life. Table 2 lists the main notation used in this section.

TABLE 2. Table of notation.

symbol	explanation
x	feature vector
y	tagged value
i	the i-th iteration
j	j-th sample
t	regression tree
r	residual of a sample
sq	square variance of a node
α	learning rate
s	disk
T	GBRT model
H	pre-warning handling strategy
Y	migration target
M	migration data volume

1) REGRESSION TREES

GBRT is an ensemble method using regression trees as weak learners; the training algorithm for regression trees is given in [12, Algorithm 2]. We use the disks' SMART attributes as input vectors together with their remaining working time as the target values. For a given SMART attribute vector $\mathbf{x}$, the regression tree maps it to a quantitative value (the output) by passing $\mathbf{x}$ down the tree: the path followed is determined by the feature values of $\mathbf{x}$ and the output is the weight of the leaf node that is reached.

Figure 3 depicts a toy example of a regression tree for predicting disk residual life; we use it to illustrate how we determine the weights of each node. Throughout this process, healthy samples are ascribed a residual life of 1000 hours (i.e., 41 days and 16 hours, despite most disks likely surviving for years). When predicting a disk's residual life, we start at the root node (node 1), which is weighted with the mean residual life (516.3 hours) of all samples. Node 1 splits according to the SMART attribute "Power On Hours": if a sample's value is $\leq$ 95, it moves to node 2 (these samples have a mean residual life of 359.1 hours), otherwise it moves to node 9 (with a mean residual life of 910.2 hours). Nodes 2 and 9 instead split according to the value of the SMART attribute "Reallocated Sectors Count (raw value)". This continues until the maximum tree depth d has been reached (in this example, we have $d = 4$).

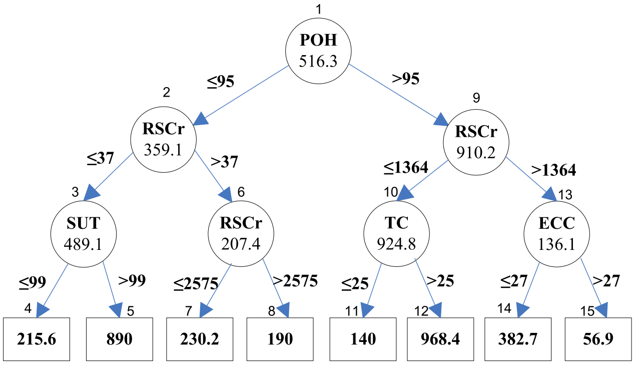


FIGURE 3. A toy example regression tree: nodes are labeled 1 through 15, and node weights (in hours) are determined by the mean residual life of disks at that node. We use the weights of the leaf nodes to predict the residual life of a disk. The SMART attributes are: "POH" = "Power On Hours", "RSCr" = "Reallocated Sectors Count (raw value)", "TC" = "Temperature Celsius", "SUT" = "Spin Up Time", "ECC" = "Hardware ECC Recovered". The theoretical maximum node weight is 1000 hours.

TABLE 3. Details of the datasets "W", "M", and "S".

Dataset	Label	# disks	# samples	Period (days)
"W"	Good	22,962	3,837,568	7
	Failed	433	158,150	20
"M"	Good	10,010	1,681,680	7
	Failed	147	79,698	25
"S"	Good	38,819	5,822,850	7
	Failed	170	97,236	25

We choose the best way to split a node (e.g. the condition POH $\leq$ 95 on node 1 in Figure 3) by minimizing

$$sq := \sum_j (y_j - \bar{y})^2, \tag{4}$$

where y_j is the remaining life of the disk in sample j at the time the sample is taken, and $\bar{y} = \mathrm{ave}_j(y_j)$. A *sample* comprises the SMART attributes of a disk at a given time point; in this work, we take samples hourly for each disk. The sum (4) is over all samples j that satisfy the splitting conditions of each ancestor node (e.g. at node 6 in Figure 3, the samples are those which satisfy both POH $\leq$ 95 and RSCr $>$ 37). This approach deviates from using the "greatest gain in information", which is ordinarily used for classification methods.

Regression trees tend to overfit when limited to data with few samples in a high-dimensional space (i.e., large numbers of features). In our data set (see Tables 3 and 4), there are millions of samples and dozens of features, implying regression trees do not encounter this problem.

2) ORIGINAL GBRT MODEL

Like other boosting methods, a GBRT is built by performing gradient descent in a function space. For $i \geq 1$, we define the model $T^{(i)}$ at the i-th iteration by

$$\begin{aligned} T^{(i)} &= T^{(i-1)} + \alpha\, t_i, \\ &= \sum_i \alpha\, t_i, \end{aligned} \tag{5}$$

TABLE 4. Basic features (i.e., SMART attributes) for the "W", "M", and "S" datasets.

ID	Feature Name	Datasets
1	"Raw Read Error Rate"	"W", "M", "S"
2	"Spin Up Time"	"W", "M", "S"
3	"Reallocated Sectors Count"	"W", "M", "S"
4	"Seek Error Rate"	"W", "M", "S"
5	"Power On Hours"	"W", "M", "S"
6	"Reported Uncorrectable Errors"	"W"
7	"High Fly Writes"	"W"
8	"Temperature Celsius"	"W", "M", "S"
9	"Hardware ECC Recovered"	"W"
10	"Current Pending Sector Count"	"W", "M", "S"
11	"Reallocated Sectors Count" (raw value)	"W"
12	"Current Pending Sector Count" (raw value)	"W"

with node-wise addition and scalar multiplication, where t_i is a regression tree and α is a learning rate defined by user. In each iteration (we use 500 iterations), the regression tree t_i is selected to minimize a loss function L for the given model $T^{(i-1)}$. This paper uses the square loss $L := \frac{1}{2}\sum_j (T^{(i)}[j] - y_j)^2$ as the loss function, where $T^{(i)}[j]$ denotes the prediction result of the j-th sample by $T^{(i)}$, and y_j is the tagged (target) value for the j-th sample, which achieves the minimum when $T^{(i)}[j] = y_j$ for all samples.

As deduced by [32, eqs. (2) and (3)] and [15], the gradient for a sample j is the residual (i.e., the prediction error) vs. the tagged value y_j from the previous iteration). Specifically, we can give the residuals from the i-th model (determining the $(i+1)$-th regression tree) by

$$\begin{aligned} r^{(i+1)}[j] &= y_j - T^{(i)}[j], \\ &= r^{(i)}[j] - \alpha\, t_i[j], \end{aligned} \tag{6}$$

where $t_i[j]$ is the prediction result for the j-th sample from the i-th tree, and $r^{(1)}[j] = y_j$, and (4) generalizes to

$$sq = \sum_j \left(r^{(i)}[j] - \overline{r^{(i)}}\right)^2. \tag{7}$$

When training the GBRT models, like in our previous work [15], we use disks' SMART attributes along with their change rates (the critical features) as the input vectors; we use the disks' residual-life times as the target values. The details of the GBRT training process are given in Algorithm 1.

3) IMPROVED GBRT MODEL

In the original GBRT algorithm, each SMART record (i.e., the collection of a disk's SMART attribute values at some time point) is treated as a single input sample, and when training, the ultima aim of the prediction is fitting the tagged value (i.e., the prediction result) of each sample to its tagged value. Therefore, at each iteration, we add a new tree to reduce the error between the tagged value and current prediction result for each sample. The original GBRT algorithm thereby only focuses on predicting the residual life for each disk based on one sample, rather than protecting at-risk data in system.

Algorithm 1 GBRT Model Training Procedure

```
Input: Training set (SMART attributes; residual life $y_j$ for
    each sample $j$), learning rate $\alpha$, tree depth $d$, number of
    regression trees $c$
Output: disk residual-life prediction model GBRT $T^{(c)}$
initialize $r^{(1)}[j] \leftarrow y_j$ for each sample $j$
for regression tree $i$ from 1 to $c$ do    ▷ build regression tree $t_i$ of depth $d$
    assign the root node of $t_i$ the weight $\overline{r^{(i)}}$
    for $k$ from 1 to $d$ do
        for each node $V$ at depth $k$ do
            for each possible split at $V$ do
                compute $sq_{\mathrm{L}} + sq_{\mathrm{R}}$ from (7), where $L$ and
                    $R$ are $V$'s proposed child nodes
            end for
            split $V$ to minimize $sq_{\mathrm{L}} + sq_{\mathrm{R}}$
            assign $V$'s child nodes with weight
                $\mathrm{ave}_s(r^{(i)}[s])$, where the average is over all
                disks $s$ which satisfy the splitting conditions
                of its ancestor nodes
        end for
    end for
    update $r^{(i+1)}[j] \leftarrow r^{(i)}[j] - \alpha\, t_i[j]$ for each sample $j$
end for
$T^{(c)} = \sum_{i=1}^{c} \alpha\, t_i$
```

In order to maximize MR, minimize MMR, and maximize MT and MMT,we modify the loss function to improve the training algorithm of GBRT model. We use migration errors instead of prediction errors as loss in the function. To this end (a) we use each training disk (rather than an individual sample) as one multiple-sample instance, i.e., the SMART samples from a single disk are treated as a whole instance; and (b) we incorporate some pre-warning handling strategies H (such as the method proposed in Section III-A) into the modified loss function.

Specially, the loss function is modified as

$$L' = \sum_s \left(Y_s - M_s(T^{(i)}, H)\right)^2,$$

where Y_s is the migration target of disk s (if s is an at-risk disk, Y_s is the capacity of s; if s is a healthy disk, $Y_s = 0$), and $M_s = M_s(T^{(i)}, H)$ is the amount of data migrated from s based on $T^{(i)}$ (the prediction from the i-th additive model) and pre-warning handling strategy H. If all the at-risk data are successfully migrated to health disks and no data on healthy disks are mismigrated (i.e., $Y_s = M_s$ for all disks), then L' achieves its theoretical minimum.

To minimize L', target residuals are modified to be calculated per disk (instead of per sample): if the disk meets its migration target based on the current prediction and the pre-warning handling strategy, the residuals of the samples from this disk are set to zero; otherwise, they are given by (6).

Specifically, for GBRT training algorithm, we change (6) so that

$$r^{(i+1)}[j] := 0 \tag{8}$$

at whatever time the existing regression trees and pre-warning handling strategy correctly handles the data for the j-th input sample (migrating those data if and only if it is necessary, and at a suitable rate).

In each iteration, a new tree is introduced to adjust for migration errors (or residuals) for every disk vs. the migration target Y from the previous prediction model (rather than adjusting according to prediction errors). That is, only those disks which do not meet their migration targets (based on the prediction from the current model) need further prediction through a new tree. By this means, the new tree focuses on the samples that are not predicted appropriately by the existing predictors (and result in improper pre-warning handling for the corresponding disks).

IV. DATASETS

A. DESCRIPTION

There are three main types of disk failures: (a) permanent whole-disk failures, where a disk stops working permanently and needs to be replaced; (b) transient performance problems, where a disk is only temporarily inaccessible and can be accessed after several connection attempts; (c) partial-disk failures, where some sectors on a disk cannot be accessed but which can be corrected by a redundancy mechanism inside the disk, e.g. error correcting codes. We focus on permanent whole-disk failures, which are the most harmful.

To validate the proposed metrics and prediction model, we collect disk SMART records from two large-scale data centers. We take hourly samples of each working disk, each containing a disk's SMART values at that time point. The statistics of the datasets are listed in Table 3.

We use "W" to denote the dataset[1] used in [9], which was collected from one data center. This dataset has 23,395 disks data from a single disk model recorded in "W", which are labeled "good" or "failed". One week of SMART samples are recorded for each good disk, and nearly 20 days worth of samples (those prior to failure) are recorded for each failed disk.

We use "M" and "S" to denote the additional two datasets[2] collected from a second data center, which were used in [16]. They contain 49,146 disks in total, including two Seagate disk models (different to the model in "W"). One week of samples were taken for each good disk, and about 25 days worth of samples before failure for each failed disk.

Each disk in "W" reports 23 SMART attributes. We filter out those which are irrelevant (i.e., they do not vary), after which there are only 10 attributes remaining. Each SMART attribute has two values: a 6-byte raw value, and a 1-byte normalized value calculated from the raw value. Since some raw values are more sensitive to disk failure prediction, we additionally choose two raw values as *basic features*. We select in total 12 basic features for the "W" dataset, which are listed in Table 4. For disks in the "S" and "M" datasets, due to the lack of information, we only choose the 7 basic features indicated in Table 4 to build prediction models.

B. DATA PREPROCESSING

To reflect how SMART samples vary over time, we incorporate changes in SMART attributes. As in the authors' previous work [12], we define a *change feature* for each basic feature: the absolute differences between the current value and its corresponding value 6 hours prior. We apply rank-sum test, arrangement test, and z-scores [4], [12] to select the *critical features* from the basic and change attributions.

For the "W" dataset, the basic features 1 through 9 and 11 (as in Table 4), together with the change features calculated from 1, 9, and 11, are selected as the critical features. For datasets "M" and "S", the selected critical features are instead features 1 through 5, 8, and 10, along with the change features for 1 and 3.

To model the online disk failure prediction (i.e., to train the prediction model using historical data), we divide the datasets into training and test sets with respect to time: for healthy disks, the training data comprises the earlier 70% of the samples, and the test data comprises the later 30%. Since the datasets didn't record the chronological order of disk failures, we randomly divide the failed disks in them into training and test sets in a 7 to 3 ratio.

Because healthy disks are far more numerous than failed disks (from 53 times to 228 times more numerous in the datasets "W", "M", and "S"), we use a restricted set of healthy samples for training the GBRT models. Specifically, for each healthy disks in the training set, we randomly select 3 samples for the dataset "W" and 1 sample for the datasets "S" and "M" as healthy training samples. The test data set remains unchanged.

We perform all experiments on a standard desktop personal computer, since they do not require significant

[1] The dataset is available from http://pan.baidu.com/share/link?shareid=189977&uk=4278294944.

[2] The datasets are available from https://github.com/nkdsliu/diskdata.

TABLE 5. Migration performance of the models CT and RNN on the "W" dataset, in terms of MR and MMR, for various migration times (equal to both MT and MMT). The row labeled "(FDR/FAR)" is equivalent to when migration is instantaneous (i.e., a 0 migration time), in which case FDR = MR and FAR = MMR.

migration time (hours)	CT		RNN	
	MR (%)	MMR (%)	MR (%)	MMR (%)
(FDR/FAR)	95.49	0.0900	98.47	0.5134
2	95.11	0.0900	98.47	0.4936
7	94.85	0.0634	98.36	0.4357
24	93.98	0.0302	96.31	0.3684
72	91.11	0.0126	92.49	0.1770
120	89.83	0.0076	90.70	0.1080
168	88.26	0.0054	88.10	0.0783
240	84.11	0.0038	81.57	0.0548
336	78.54	0.0027	73.79	0.0392

computational resources. Training the GBRT model finishes within ten minutes, and residual-life prediction is performed for nearly 10,000 disks per second. Thus, the overhead of the modified GBRT method is unproblematic for online and real-time running in large-scale and cloud storage systems.

V. EXPERIMENTAL RESULTS

A. METRIC COMPARISON

Here, we describe how to evaluate the performance of disk-failure prediction models (both binary classification and residual-life prediction) using the proposed new metrics (evaluating migration accuracy and migration time) instead of the previous metrics (evaluating classification accuracy). We also discuss the trade-off between the two groups of new metrics. The "W" dataset is used in these experiments.

1) BINARY CLASSIFICATION MODEL

This experiment gives an example demonstration of how we envisage evaluating binary classification models using MR, MMR, MT, and MMT, and compare them to the traditional metrics FDR, FAR, and TIA.

The research [12], [16] indicated that the classification tree (CT) model and recurrent neural network (RNN) model outperform other binary classification models at predicting disk failure. We evaluate the performance of these models using new metrics of MR, MMR, MT, and MMT, along with traditional metrics of FDR, FAR, and TIA, and adopt the practices in [12] and [16] to preprocess data and build the CT and RNN models, respectively.

When training the CT model, we set the time window to 168 hours: the last 168 samples prior to actual failure per failed disks in the training set are used as failed training samples. We describe the preprocessing of healthy disks in the training set in Section IV-B. When detecting failures, we use a simple detection method where we predict a disk is about to fail if the model classifies any one sample as failed. We observe that the CT model reaches a FDR of 95.49% at 0.09% FAR, with a TIA of 354.6 hours [12, Table 4]. When testing RNN model, we likewise use this simple detection method, and the RNN model attains a FDR of 98.47% at 0.51% FAR, with a TIA of 294.0 hours.

To test the models, we sequentially process the samples of each disk in the test set. If a disk is detected to fail, its data is be migrated to other healthy disks at a transfer rate, until the disk fails or its data has been completely migrated, while measuring the models' performance on MR and MMR. Since the prediction results are binary values (indicating the disk is or not about to fail), the *migration transfer rate* is fixed for all pre-warning migrations, and, for an m TB disk, we set it to one of $\{m/2, m/7, m/24, m/72, m/120, m/168, m/240, m/336\}$ TB/h, so consequently the migration time from an at-risk disk (MT and MMT) is one of {2, 7, 24, 72, 120, 168, 240, 336} hours accordingly.

Table 5 lists the migration performance (measured by MR and MMR) of the CT and RNN models as binary classification models as the migration times (equal to MT and MMT) vary. We see the migration accuracy of both models changes as the migration time varies, which indicates that MR and MMR vary with MT and MMT.

The metrics FDR and FAR are also included in Table 5, which is equivalent to MR and MMR respectively, when all at-risk data are migrated successfully. When the migration time is 2 hours for the CT model and ≤ 7 hours for the RNN model, MR is close to the FDR (i.e., approximately all data are successfully migrated after a failure prediction). However, such low migration times are achieved at the expense of high network and disk bandwidth consumption, reducing the quality of service, especially during simultaneous failures.

Practically, cloud and large storage systems are usually unable to assign such sufficient resources to ensure all pre-warning migrations are successful; they can only afford a relatively low migration bandwidth (implying a long MT and MMT). As such, there will be some at-risk data can not be completely migrated before failure, although they are successfully detected by prediction model, as per Table 5 where we see that MR deteriorates as MT increases.

In addition, we observe that the RNN model outperforms CT model in terms of FDR, which implies that the RNN

TABLE 6. Performance of the GBRT and RNN models on the "W" dataset in terms of MR, MMR, MT, MMT, ACC_h, and ACC_f.

Model	Previous Metrics		New Metrics			
	ACC_f (%)	ACC_h (%)	MR (%)	MMR (%)	MT (h)	MMT(h)
RNN	30.28	99.333	78.54	0.0809	258.37	311.90
GBRT	23.69	99.987	86.50	0.0029	107.55	289.74

model predicts at-risk disks more accurately. However, with a migration time of $\geq$ 168 hours, the CT model achieves a better MR, that implies it could protect (i.e., successfully migrate) more data from the at-risk disks. Importantly, in some cases migration is not completed in sufficient time, such as within a 168-hour window, as we use in this paper. Thus, the previous metrics (i.e., FDR and FAR) evaluating prediction accuracy have misleading results: the RNN model significantly outperforms the CT model (in terms of FDR), despite successfully migrating (and thus protecting) less at-risk data.

In comparison with the previous metrics (i.e., FDR and FAR), the new proposed metrics (i.e., MR and MMR along with MT and MMT) give system operators a more clear and realistic indication of how a binary classification model protects data (with certain resource consumption) for cloud and large storage systems.

When testing the models, we have experimented the voting-based detection method as used in [12] and [16], and the results are similar to Table 5; we omit the full details.

2) RESIDUAL-LIFE PREDICTION MODEL

This experiment gives an example demonstration of how we envisage comparing residual-life predictors using the new proposed metrics, MR, MMR, MT, and MMT, vs. the previous metric ACC.

The research [16] also indicated the RNN model outperforms other prediction models in term of ACC at predicting disk health status. Thus, the RNN model is used as the control group to evaluate the proposed GBRT model in disk residual-life prediction issues. And, we adopt the practices in [16] to preprocess data and build the RNN prediction models.

When training the GBRT models, to reduce mismigration, the residual-life interval of healthy disks is adjusted from "$>$ 500 hours" to "$>$ 800 hours". However, when testing the GBRT and RNN models, the residual life of 500 hours continues to be used as the boundary between healthy and failed.

When testing the residual-life predictors, we sequentially process the samples of each disk in the test set, and use the pre-warning handling strategy listed in Table 1. If the predicted residual life of a disk is in level 6, then no pre-warning migration is performed. Otherwise, its data is migrated to other healthy disks at the rate specified in Table 1, until its data has been completely migrated, while measuring the models' performance on MR and MMR. Meanwhile, we record the migration time for each disk that is predicted to fail, measuring the MT and MMT. For a partially migrated disk, which fails before migration is complete, we scale its actual process time to the expected completion time. Unless otherwise stated, we use the same pre-warning handling strategy in the following experiments.

Moreover, for healthy disks, we compute the proportion of healthy samples which are predicted into level 6 as the value of ACC_h. And for failed disk, we compute the proportion of failed samples which are predicted into the right intervals (levels 1 through 5), as the value of ACC_f.

The residual-life intervals in Table 1 are not equal, so an imbalanced number of training samples fall into different intervals, which may negatively impact GBRT-model predictions. Therefore, for every failed disk in the training set, we select two SMART records from each interval as training samples to train the GBRT models (unless otherwise stated below). When building the GBRT models, we set the following parameters: learning rate $\alpha = 0.1$, tree-depth $d = 4$, and number of iterations $c = 500$.

Table 6 lists the performance of the GBRT and RNN models as disk residual-life predictors, in terms of MR, MMR, MT, MMT, ACC_h, and ACC_f.

We observe that the RNN model has a higher ACC_f than the GBRT model, i.e., the RNN model more frequently predicts the failed samples within the correct urgent level. However, the GBRT model has a better MR, i.e., the GBRT model successfully migrates more data from at-risk disks. The GBRT model outperforms the RNN model on what actually matters: protecting data. We attribute this behavior to how most samples that are correctly predicted by the RNN model are less urgent (mostly levels 5 or 4) with a low migration rate, resulting in RNN having an inflated ACC_f.

Importantly, this is an example of how the previous metric (ACC_f or ACC_h), gives misleading results: the RNN model significantly outperforms the GBRT model (in terms of ACC_f), despite successfully migrating (and thus protecting) less at-risk data.

The new evaluation metrics (MR, MMR, MT, and MMT), which measure a prediction model's ability to protect data and bandwidth cost, are more naturally meaningful than the previous metrics (ACC_f and ACC_h). The implications of the trade-off between ACC_f and ACC_h are not apparent, whereas MR/MT directly indicates how much we successfully migrate and MMR/MMT directly indicates how much we waste.

Moreover, since the new metrics measure the actual results of the failure prediction models, we can also use them to compare the performance of binary classification models against residual-life prediction models, which is not possible using the previous metrics.

TABLE 7. Migration performance of the GBRT model on the "W" dataset as the migration time varies.

k	MR (%)	MMR (%)	MT (h)	MMT (h)
0.8	85.40	0.0023	134.13	362.18
1.0	86.50	0.0029	107.55	289.74
1.2	87.25	0.0035	89.74	241.45
1.4	87.77	0.0041	76.88	206.96
1.6	88.17	0.0047	67.50	181.09

TABLE 8. Migration performance of the original and improved GBRT models on the "W" dataset. "GBRT+" denotes the modified GBRT model.

Model	MR (%)	MMR (%)	MT (h)	MMT (h)
GBRT	86.50	0.0029	107.55	289.74
GBRT+	88.80	0.0013	124.58	329.36

3) TRADE-OFF BETWEEN THE NEW METRICS

Using more bandwidth for pre-warning handling prior to an incomplete migration implies that more of the data is protected. Consequently, the shorter the migration time (MT and MMT), the higher the migration rate (MR and MMR).

In this section, we observe how the migration performance of GBRT model changes as the migration time varies. We use the partition for disk residual life in Table 1, and change the migration time by multiplying the migration transfer rate by $k \in \{0.8, 1.0, 1.2, 1.4, 1.6\}$. As k increases, the migration transfer rate increases, and thus the migration time decreases. The results are listed in Table 7.

As migration time decreases (thereby increasing resource consumption), the GBRT model's ability to protect data improves, leading to a trade-off between data protection and resource consumption, and system operators may adjust the migration rates to achieve suitable levels of reliability and availability. Moreover, we conclude that the metrics MR and MMR alone are incapable of giving a comprehensive evaluation for disk failure prediction models, as they do not incorporate resource costs.

B. EVALUATING THE IMPROVED GBRT ALGORITHM

In Section III-C.3 we propose a modification (8) to the target residual (error) computation which improves residual life prediction; we call this method GBRT+. In each iteration, we update the residuals of all samples from a single disk together: if the disk is handled correctly based on the current prediction and pre-warning handling strategy, the target residuals are set to 0; otherwise, the target residuals are set to the prediction errors vs. the target value from the current prediction by (6).

We use the "W" dataset in the experiments in this section. When building the GBRT+ model, we set the learning rate α to 0.12. In Table 8 we list the experimental results.

As expected, the GBRT+ model has better migration accuracy and less resource costs than the original GBRT model. While the improvements are relatively minor, the cloud and large-scale storage systems makes even minor improvements in migration accuracy and resource costs worthwhile.

TABLE 9. Migration performance of the GBRT+ model on the "M" and "S" datasets.

Dataset	MR (%)	MMR (%)	MT (h)	MMT (h)
"M"	92.92	0.0059	179.73	245.01
"S"	93.89	0.0022	173.41	297.82

TABLE 10. Migration performance of the GBRT+ model on small-sized synthesized datasets.

Dataset	MR (%)	MMR (%)	MT (h)	MMT (h)
W_1	87.15	0.0051	115.66	144.00
W_2	88.82	0.0024	82.03	273.46
W_3	91.34	0.0011	125.86	326.66
W_4	84.17	0.0016	133.58	321.62

C. VERIFYING PRACTICAL USABILITY

We want to verify the practical usability of GBRT+ by simulating its application in a real-world data center, which may involve various disk families or multiple disk models, and we also investigate its use for small-scale data centers.

1) PERFORMANCE ON "M" AND "S" DATASETS

The different characteristics of various disk models (even those by the same manufacturers) potentially affect their reliability. Consequently, it is important to verify that hard-disk failure prediction models remain effective over varying disk models. For this reason, we evaluate the improved model GBRT+ on the "M" and "S" datasets, which have different disk models from those of the "W" dataset. For every failed disk in the training sets, we select three samples from each residual-life interval, as the training samples to train the GBRT+ model. In Table 9 we list the results of these experiments.

On the "M" and "S" datasets, we observe that GBRT+ maintains comparable performance with that on the "W" dataset, on both migration accuracy (in terms of MR and MMR) and resource cost (in terms of MT and MMT). The experimental results verify the effectiveness of our proposed GBRT+ model with different disk models.

2) PERFORMANCE ON SMALL DATASETS

The datasets "W", "M", and "S" each have a large number of hard disks, and, while disk failure prediction models are also used in small and medium-sized data centers. In order to verify the effectiveness of the improved GBRT+ model in these environments, we evaluate it using four small synthesized datasets, denoted W_1, W_2, W_3, and W_4, by randomly choosing 10%, 25%, 50%, and 75% respectively from all the disks (both healthy and failed) from the "W" dataset. The smallest dataset W_1 has only 2,296 healthy disks and 43 failed disks. We list the experimental results in Table 10. We observe that with all the four datasets, the improved GBRT+ model achieves acceptable migration performance.

3) PERFORMANCE ON HYBRID DATASET

As a data center grows, it may be cost effective to use multiple distinct disk models, and this situation is not uncommon in

TABLE 11. Migration performance of the GBRT+ model on the "MS" dataset.

MR (%)	MMR (%)	MT (h)	MMT (h)
95.34	0.0047	194.23	296.29

real-world data centers. Building a prediction model for each possible disk model is impractical, so it is necessary to use samples from multiple disk models to train failure prediction models. Thus, we merge the "M" and "S" datasets to generate a hybrid dataset (which we denote "MS"). For every failed disk in training set, three samples from each residual-life interval are selected as the training samples to train the GBRT+ model. In Table 11 we list the performance of GBRT+ on the "MS" dataset; we see this setup is also practicable.

VI. CONCLUSIONS

This paper contend that the existing metrics (i.e., FDR, FAR, and ACC) used for evaluating disk failure prediction models are inadequate for comparing and selecting models, particularly for cloud and large storage systems.

To address these limitations, we present:

- migration rate (MR) and mismigration rate (MMR), as proposed by some of the present authors in [18], which measure how much at-risk data is protected (successfully migrated) and how much data is unnecessarily protected, respectively; and
- migration time (MT) and mismigration time (MMT), which measure how long it takes to complete migration for an at-risk disk and how long it takes to complete the migration for a false alarm, respectively.

MT and MMT reflect the mean migration transfer rate and are used to measure the consumption of resources used in protecting at-risk data.

We compare the proposed GBRT model and the RNN model (as disk residual-life prediction models) and encounter an undesirable property: the RNN model achieves better prediction performance (and thus has better ACC) but successfully migrates less data from at-risk disks (worse MR) and unnecessarily migrates more data from healthy disks (worse MMR). It is therefore misleading to only compare the models using ACC: we are prematurely declaring migrations as successful before the data are actually migrated. This becomes even more problematic when we note that prediction models might be designed to optimize ACC, or some other prediction accuracy metric.

The models may have different performance on MR and MMR as migration time (MT and MMT) varies. That is, MT and MMT describe the resource cost to achieve the certain MR and MMR. Thus, these two groups of metrics should be used together to measure the practical usage of disk failure prediction models in cloud storage systems.

While there are known methods for comparing the accuracy of prognostic metrics (such as [33]), this paper changes the criterion from "accuracy" to "resource cost" (in terms of unnecessary migration). For the problem of disk-failure prediction, one of the main messages of this paper is that "accuracy" and "cost" do not have a simple relationship: greater accuracy does not necessary imply less resource cost, an assumption which is inherent in e.g. (1). Contributing to this complicated accuracy-cost relationship, we incorporate dynamic forms of maintenance (varying migration rates based on urgency) and incomplete maintenance (i.e., incomplete migration), both of which entail different predictions incurring different resource costs.

The proposed GBRT+ model predicts the residual life for each disk, enabling system operators to migrate data from the at-risk disks based on their urgency, which can ensure both the reliability and the availability of storage systems. Moreover, in cloud computing platforms, such as Microsoft Azure, ranking disks according to their residual life (or error-proneness) [30] can help the service systems to allocate a virtual machine to a much healthier one, therefore improving service availability. Experimental results show that the GBRT and GBRT+ models are useful and applicable to real-world data centers.

Among those metrics listed by Saxena *et al.* [19], two stand out as promising candidates for capturing the notion of accuracy while approaching the point of failure: "timeliness" and "convergence". Both of these metrics weigh more heavily inaccurate predictions close to the time of failure. As a future research direction, we propose using these two metrics for urgency-weighted evaluations of disk-failure prediction models. Also with this motivation, we may adjust the machine learning process to consider poor predictions nearing actual failure more severe.

ACKNOWLEDGEMENTS

The authors would like to thank the anonymous referees of this article for their valuable feedback, particularly for drawing to our attention the relevant prognostics literature which has helped place this work in a broader context.

REFERENCES

[1] Q. Xin, E. L. Miller, and S. J. T. J. E. Schwarz, "Evaluation of distributed recovery in large-scale storage systems," in *Proc. IEEE HPDC*, Jun. 2004, pp. 172–181.

[2] F. Mahdisoltani, I. Stefanovici, and B. Schroeder, "Proactive error prediction to improve storage system reliability," in *Proc. USENIX ATC*, 2017, pp. 391–402.

[3] J. Xiao, Z. Xiong, S. Wu, Y. Yi, H. Jin, and K. Hu, "Disk failure prediction in data centers via online learning," in *Proc. ICPP*, 2018, pp. 1–10.

[4] J. F. Murray, G. F. Hughes, and K. Kreutz-Delgado, "Machine learning methods for predicting failures in hard drives: A multiple-instance application," *J. Mach. Learn. Res.*, vol. 6, pp. 783–816, May 2005.

[5] G. Hamerly and C. Elkan, "Bayesian approaches to failure prediction for disk drives," in *Proc. Conf. Mach. Learn.*, 2001, pp. 202–209.

[6] G. F. Hughes, J. F. Murray, K. Kreutz-Delgado, and C. Elkan, "Improved disk-drive failure warnings," *IEEE Trans. Rel.*, vol. 51, no. 3, pp. 350–357, Sep. 2002.

[7] J. F. Murray, G. F. Hughes, and K. Kreutz-Delgado, "Hard drive failure prediction using non-parametric statistical methods," in *Proc. Artif. Neural Netw.*, 2003, pp. 1–4.

[8] Y. Zhao, X. Liu, S. Gan, and W. Zheng, "Predicting disk failures with HMM- and HSMM-based approaches," in *Proc. Ind. Conf. Data Mining Adv. Data Mining. Appl. Theor. Aspects*. Berlin, Germany: Springer, 2010, pp. 390–404.

[9] B. Zhu, G. Wang, X. Liu, D. Hu, S. Lin, and J. Ma, "Proactive drive failure prediction for large scale storage systems," in *Proc. MSST*, May 2013, pp. 1–5.

[10] Y. Wang, Q. Miao, and M. Pecht, "Health monitoring of hard disk drive based on Mahalanobis distance," in *Proc. Prognostics Syst. Health Manage. Conf.*, 2011, pp. 1–8.

[11] Y. Wang, Q. Miao, E. W. M. Ma, K.-L. Tsui, and M. G. Pecht, "Online anomaly detection for hard disk drives based on Mahalanobis distance," *IEEE Trans. Rel.*, vol. 62, no. 1, pp. 136–145, Mar. 2013.

[12] J. Li *et al.*, "Hard drive failure prediction using classification and regression trees," in *Proc. 44th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw.*, Jun. 2014, pp. 383–394.

[13] A. Ma, F. Douglis, G. Lu, D. Sawyer, S. Chandra, and W. Hsu, "RAIDShield: characterizing, monitoring, and proactively protecting against disk failures," in *Proc. USENIX FAST*, 2015, pp. 241–256.

[14] S. Wu, H. Jiang, B. Mao, "Proactive data migration for improved storage availability in large-scale data centers," *IEEE Trans. Comput.*, vol. 64, no. 8, pp. 2637–2651, 2015.

[15] J. Li, R. J. Stones, G. Wang, X. Liu, Z. Li, M. Xu, "Hard drive failure prediction using decision trees," *Rel. Eng. Syst. Saf.*, vol. 164, pp. 55–65, Aug. 2017.

[16] C. Xu, G. Wang, X. Liu, D. Guo, and T.-Y. Liu, "Health status assessment and failure prediction for hard drives with recurrent neural networks," *IEEE Trans. Comput.*, vol. 65, no. 11, pp. 3502–3508, Nov. 2016.

[17] S. Pang, Y. Jia, R. Stones, X. Liu, and G. Wang, "A combined Bayesian network method for predicting drive failure times from SMART attributes," in *Proc. IJCNN*, Jul. 2016, pp. 4850–4856.

[18] J. Li, R. J. Stones, G. Wang, Z. Li, X. Liu, K. Xiao, "Being accurate is not enough: New metrics for disk failure prediction," in *Proc. SRDS*, Sep. 2016, pp. 71–80.

[19] A. Saxena *et al.*, "Metrics for evaluating performance of prognostic techniques," in *Proc. ICPHM*, Oct. 2008, pp. 1–17.

[20] A. Saxena, J. Celaya, B. Saha, S. Saha, and K. Goebel, "Evaluating algorithm performance metrics tailored for prognostics," in *Proc. Aerosp. Conf.*, Mar. 2009, pp. 1–13.

[21] K. Goebel, A. Saxena, S. Saha, and J. Celaya, "Prognostic performance metrics," *Mach. Learn. Knowl. Discovery Eng. Syst. Health Manage.*, pp. 147–177, 2011.

[22] J. E. Dzakowic and G. S. Valentine, "Advanced techniques for the verification and validation of prognostics & health management capabilities," in *Proc. MFPT*, 2007, pp. 1–11.

[23] L. P. Queiroz *et al.*, "A fault detection method for hard disk drives based on mixture of Gaussians and nonparametric statistics," *IEEE Trans. Ind. Informat.*, vol. 13, no. 2, pp. 542–550, Apr. 2016.

[24] C. A. C. Rincón, J.-F. Pâris, R. Vilalta, A. M. K. Cheng, and D. D. E. Long, "Disk failure prediction in heterogeneous environments," in *Proc. SPECTS*, Jul. 2017, pp. 1–7.

[25] 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. FAST*, 2007, pp. 1–13.

[26] B. P. Leao, T. Yoneyama, G. C. Rocha, and K. T. Fitzgibbon, "Prognostics performance metrics and their relation to requirements, design, verification and cost-benefit," in *Proc. PHM*, Oct. 2008, pp. 1–8.

[27] A. Qin, D. Hu, J. Liu, W. Yang, D. Tan, "Fatman: Cost-saving and reliable archival storage based on volunteer resources," *Proc. VLDB Endowment*, vol. 7, no. 13, pp. 1748–1753, 2014.

[28] X. Ji *et al.*, "A proactive fault tolerance scheme for large scale storage systems," in *Algorithms and Architectures for Parallel Processing*. Cham, Switzerland: Springer, 2015, pp. 337–350.

[29] P. Li, J. Li, R. J. Stones, G. Wang, Z. Li, and X. Liu, "ProCode: A proactive erasure coding scheme for cloud storage systems," in *Proc. SRDS*, Sep. 2016, pp. 219–228.

[30] Y. Xu *et al.*, "Improving service availability of cloud systems by predicting disk error," in *Proc. USENIX ATC*, 2018, pp. 481–494.

[31] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, 2001.

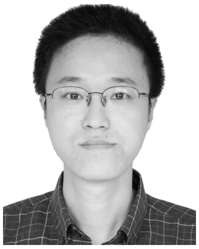
[32] A. Mohan, Z. Chen, K. Weinberger, Web-search ranking with initialized gradient boosted regression trees," in *Proc. JMLR, Workshop Conf.*, 2011, pp. 77–89.

[33] B. P. Leão, J. P. P. Gomes, R. K. H. Galvão, and T. Yoneyama, "How to tell the good from the bad in failure prognostics methods," in *Proc. Aerosp. Conf.*, Mar. 2010, pp. 1–7.

JING LI received the B.Sc. and M.Sc. degrees in computer science and technology from Shandong University, Jinan, China, in 2004 and 2007, respectively, and the Ph.D. degree in computer science from Nankai University, Tianjin, China, in 2016.

She is currently a Teacher at the College of Computer Science and Technology, Civil Aviation University of China. Her research interests include mass data storage and machine learning.

REBECCA J. STONES received the Ph.D. degree in pure mathematics from Monash University in 2010. She is currently an Associate Professor at Nankai University. She currently has diverse research interests, including combinatorics and graph theory, codes and cryptography, search engines and data storage, phylogenetics, and quantitative psychology.



GANG WANG received the B.Sc., M.Sc., and Ph.D. degrees in computer science from Nankai University, Tianjin, China, in 1996, 1999, and 2002, respectively. He is currently a Professor at the College of Computer and Control Engineering, Nankai University. His research interests include storage systems and parallel computing.

ZHONGWEI LI received the Ph.D. degree in computer science and technology from Harbin Engineering University, Harbin, China, in 2006. He is currently an Associate Professor at the College of Software, Nankai University, Tianjin, China. His research interests include machine learning and mass data storage.

XIAOGUANG LIU received the B.Sc., M.Sc., and Ph.D. degrees in computer science from Nankai University, Tianjin, China, in 1996, 1999, and 2002, respectively. He is currently a Professor in computer science at Nankai University. His research interests include parallel computing and storage system.

JIANLI DING received the Ph.D. degree in operational research and cybernetics from Nankai University, Tianjin, China, in 2004. He is currently a Professor at the College of Computer Science and Technology, Civil Aviation University of China. His research interests include risk monitoring and control, and reliability analysis.