Discovering Rules from Disk Events for Predicting Hard Drive Failures

Vipul Agrawal, Chiranjib Bhattacharyya
Indian Institute of Science
Bangalore, India
{agrvipul,chiru}@csa.iisc.ernet.in

Thirumale Niranjan, Sai Susarla
NetApp Advanced Technology Group
Bangalore, India
{niranjan.thirumale,sai.susarla}@netapp.com

***Abstract*—Detecting impending failure of hard disks is an important prediction task which might help computer systems to prevent loss of data and performance degradation. Currently most of the hard drive vendors support self-monitoring, analysis and reporting technology (SMART) which are often considered unreliable for such tasks. The problem of finding alternatives to SMART for predicting disk failure is an area of active research. In this paper, we consider events recorded from live disks and show that it is possible to construct decision support systems which can detect such failures. It is desired that any such prediction methodology should have high accuracy and ease of interpretability. Black box models can deliver highly accurate solutions but do not provide an understanding of events which explains the decision given by it. To this end we explore rule based classifiers for predicting hard disk failures from various disk events. We show that it is possible to learn easy to understand rules, from disk events, which have extremely low false alarm rates on real world data.**

***Keywords*-Hard drive failures; SMART; False alarm rate; Rule-based learning;**

I. Introduction

Predicting accurately the impending failure of hard disks in the field can help systems at large organizations, data-centers and other crucial places to take corrective actions before the failure to avoid loss of data and performance degradation. Application of machine learning techniques for predicting hard disk failures are very few [1], [2], [3], [4]. The problem poses many hard challenges. It is still a matter of research as to what factors should be taken in to consideration while studying disk failures. Also, the data contains proprietary information and not freely available and thus, is difficult to acquire. Currently most of the hard drive vendors support self-monitoring, analysis and reporting technology (SMART), which measures drive characteristics such as temperature, spin-up time and data error rates. However, it has been found that SMART parameters alone are not enough to reliably predict individual drive failures [5]. Also, SMART is not correctly implemented on most systems due to lack of standards for handling of SMART data. So, we used disk events and errors generated by various layers in the storage and file system stack, referred to as ”disk events” in the rest of the paper for effective prediction. These include read and write errors, checksum errors, RAID-level errors, and others. We also included the disk model, manufacturer, disk size and the interface as our attributes. It has also been shown that there were significant variations in the occurrence of disk events across different manufacturers and even in different models by the same manufacturer [6], [7]. Also, they varied with the type of interface, i.e. FC or SATA, used in the disks.

In addition to predicting the failure of hard drives, it is also necessary to keep the false alarm rate(FAR) to a minimum. Generally, if a disk becomes erratic or its performance degrades significantly, it is offlined and thoroughly tested. However, these tests are intrusive and time-consuming, and are best kept to a minimum and applied as a last resort. Using the approach described in this paper, we can develop a system that can serve as a pre-filtering mechanism to drastically reduce the number of disks actually subjected to time-consuming diagnostics.

In this paper, we describe an application of an existing machine learning approach, MLRules algorithm [8], which is based on the rule induction principle that can be effectively employed to detect impending disk failures. The MLRules algorithm generates an ensemble classifier using decision rules as its base classifier. A simple decision rule classifier contains a set of logical statements or conditions which, when satisfied, votes for a particular class otherwise abstains from voting.

Hard drive manufacturers and vendors always need a failure prediciton technique that is easily implementable and trustworthy. Rule learning is thus a viable option. Rules can be learnt and calculated with low memory and computational requirements which is not the case with other machine learning techniques. These can be used to develop an efficient stand-alone system for predicting hard disk failures with high accuracy and low false alarm rate.

Further, rule-based techniques have special significance in case of hard disk failure prediction. They are highly interpretable i.e. intuitively comprehensible and easy to understand. Such an interpretation is not possible in case of other techniques. The interpretation can also help in pruning insignificant rules from the classification model. Rules can also provide an insight on the disk events that can be helpful in predicting failures. Such information is useful in getting to the real cause of failures in hard disks, thus improving the reliability of storage systems.

II. Related Work

Predicting failures, much before they actually occur, is necessary to allow users or storage systems enough time to backup their data, otherwise, the whole RAID reconstruction needs to be performed. Hard drive manufacturers are continuously developing self-monitoring technology in their products for such prediction. Presently, hard drives use a very naive threshold algorithm which triggers a SMART flag when any attribute exceeds a predefined value. Thresholds are set so as to avoid false alarms at the expense of predictive accuracy. They are based only on prior observations and may not have statistical basis.

Pinheiro et al. [5] found very little correlation between failure rates and elevated temperature or activity levels. They show that some SMART parameters (scan errors, reallocation counts, offline reallocation counts, and probational counts) can be helpful in predicting disk failure. Although, due to the lack of occurrence of predictive SMART signals on a large fraction of failed drives, they concluded that SMART alone cannot be used to form an accurate predictive failure model.

Bairavasundaram et al.[6], [7] found that SATA disks and their adapters develop checksum mismatches an order of magnitude more often than FC disks. They also observed that the probability of developing checksum mismatches varies significantly across different disk models even within the same disk class. Also, the fraction of disks with latent sector errors varies significantly across manufacturers and disk models. They observed that as disk size increases, the fraction of disks with latent sector errors increases across all disk models.

Hamerly and Elkan [3] applied two Bayesian methods to predict disk drive failures based on measurements of drive internal conditions. They posed it as the problem of anomaly detection. They applied a mixture model of naive Bayes clusters that is trained using expectation-maximization (NBEM). The second method was a naive Bayes classifier, a supervised learning approach. The dataset consisted of 1943 drives of which only 9 were marked as "will-fail". It achieved failed-disk detection rates of 35-40% for NBEM and 55% for naive Bayes classifier with low FAR.

Hughes et al. [2] proposed Wilcoxon rank sum statistical tests to improve failure warning accuracy and lower false alarm rates. It used Multivariate rank sum along with ORed rank sum and ORed threshold. Experimental data sets were obtained from drive design testing of 3744 drives of two different drive models with each set contains 2-3 months of reliability design test data. There were 36 verified drive failures. The highest accuracies achieved were modest (40%-60%) at 5% FAR.

Murray et al. [4] compared the performance of support vector machines(SVMs) [9], unsupervised clustering, and non-parametric statistical tests (rank-sum and reverse arrangements). Murray et al. [1] proposed an algorithm based on the multiple-instance learning framework and the naive Bayesian classifier (mi-NB). Data from 369 drives was collected, and each drive was labeled good or failed. Drives labeled as good were from a reliability demonstration test, run in a controlled environment by the manufacturer whereas the ones labeled as failed were returned to the manufacturer from users after a failure. There were only 191 failed disks in the dataset. It showed about 50% detection rates at 0% false alarm rates.

III. Rule-based Learning

Decision rules are logical statements of the form, *if condition then response*. There are many rule learning procedures [8], [10], [11], [12] to learn a compendium of rules from empirical data. A comparative review of these approaches is beyond the scope of the paper. Instead we present a short description of the MLRules procedure [8] which is a state of the art rule algorithm used in this paper.

A. MLRules Algorithm

MLRules stands for Maximum Likelihood Rules. The main idea of MLRules algorithm consists of inducing rules by greedily minimizing the negative log-likelihood to estimate the conditional class probability distribution $P(y|x)$ by which we can measure the prediction confidence. The model for the probabilities are given by a logistic like function

$$P(y=j|x) = \frac{e^{f_j(x)}}{\sum_{k=1}^{K} e^{f_k(x)}} \; \forall j \in \{1,\dots,K\} \qquad (1)$$

The problem of learning rules can then be reduced to minimizing the the following function by a MLE procedure

$$\sum_i L(y_i, \mathrm{f}(x_i)) = log\left(\sum_{k=1}^{K} e^{f_k(x_i)}\right) - f_{y_i}(x_i) \qquad (2)$$

The classification function is a linear combination of M rules

$$\mathrm{f}(x) = \sum_{m=1}^{M} r_m(x) \qquad (3)$$

IV. Deriving Rules from Disk events

In this section we discuss the application of MLRules algorithm for computing a rule based classifier from disk events.

A. Data Collection

The storage system at Netapp has a inbuilt, low overhead mechanism called Autosupport to log important system events back to a central repository. This repository (the "NetApp Autosupport Database") has also been used in studies related to disk failure [13], latent sector error [6] and data corruption [7]. The database contains logs for about four thousand different events. Learning which subset is best for prediction is itself a challenging problem and is beyond the scope of this paper. However, on the basis of domain knowledge, we selected few disk events such as media errors, checksum errors, parity errors etc. We also carried out a rigorous search on other potential events and selected those which had a significant number of occurrences for the hard disks in consideration. In view of observations in Bairavasundaram et al. [6], [7], we also added the details of disks such as disk model, disk size, manufacturer and the interface.

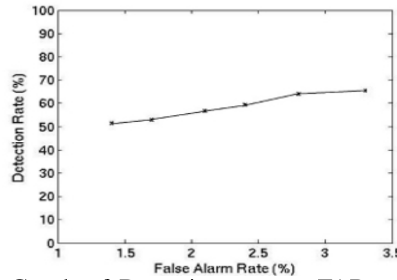


Figure 1: Graph of Detection rate vs FAR using MLRules algorithm. The points represent number of rules generated (15, 20, 30, 40, 50, 100 from left to right)

B. Dataset and Experiments

For our evaluation, we extracted the disk events data of 28877 disks returned from the field due to complaints from the customer. Out of these, 25048 were found to be good after the in-house tests and 3828 failed the test. For the purpose of creating a dataset for use with MLRules algorithm, we calculated the cumulative disk events for every disk that occurred in its lifetime. We created a matrix of dimension N x M, where N is the number of disks in the dataset and M is the number disk events under consideration. The matrix thus formed was such that each row contained a particular disk's data with disk events forming the columns.

In spite of considering 75 different types of disk events, most of the disks had relatively very few events, leaving the data set quite sparse. Most of the prevalent machine learning techniques failed to give even modest results. All the previous work regarding hard disk failure prediction have been on relatively very small datasets consisting of a few hundred failed disks. The data collected was not only field data but consisted of data collected during testing in uniform controlled environment using the SMART features. For our data we relied on the field data that was collected when the disks were in actual use. The attributes were also very different consisting of read and write errors, checksum errors, RAID-level errors and also the disk model, manufacturer, disk size etc. It did not consist of SMART attributes like temperature, flyheight etc.

Murray et al. [1] used SVM with Multiple-Instance(MI) learning technique. We cannot apply MI technique on our dataset firstly due to its larger size (about order of 2 times the magnitude) and secondly it not being time-series data. Even if we make our data a time series, the resulting dataset would be too big to make SVMs inappropriate for this kind of problem domain which prefers minimal computational and memory requirements.

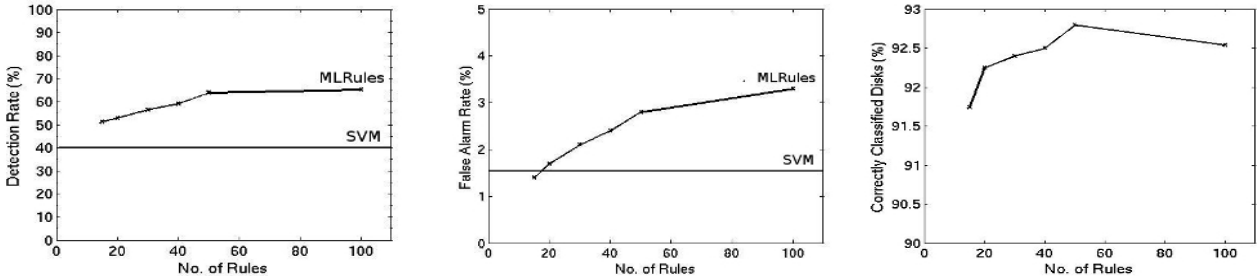
For the purpose of comparison, we applied MLRules algorithm to the dataset used in Murray et al. [1] by calculating the cumulative sum of attributes such as I/O errors and leaving out attributes such as temperature and fly height.

C. Results

For evaluation, we applied 10 fold cross-validation averaged over 10 iterations. The results are shown in Figure 1. Detection rate is defined as the number of failed disks that were classified correctly as failing. False Alarm Rate(FAR) is defined as the number of good disks that were classified as failing. Using the MLRules algorithm, we could predict the failure of around 66% of the total number of disks (28877) with only 3% FAR by generating 100 rules. If smaller values of FAR are necessary, then one need to compromise on the detection rate. The variation in detection rate and FAR can be brought about by changing the number of rules generated as shown in Figure 1. For example, detection rate falls to 60% at 2.5% FAR using 40 rules and 51% at 1.4% FAR using mere 15 rules.

The detection rate varies with the number of rules generated as shown in Figure 2a. Initially, the detection rate increases with increase in number of rules. This is due to the fact that the additional rules generated provide better identification of failed disks. Although, the false alarm rate steadily increases with number of rules as shown in Figure 2b due to increased stress on identification of failed disks that leads to good disks getting classified as failed. Considering the percentage of all disks, both failed and good, being correctly classified as shown in Figure 2c, we find that the performance steadily increases and then begins to degrade due to overfitting. It also indicates that a relatively small number of rules are sufficient to get the best results for such a system.

To benchmark the MLRules algorithm against other classifiers, we implemented a SVM with gaussian kernel. We used cross validation to find the appropriate parameters. We achieved detection rates of range 20-40% only at 1.5-2.5% FAR. It though can achieve a good FAR but the detection rate was not satisfactory. We feel that to obtain comparable accuracy one probably needs to design kernels which are more suited for this purpose than the gaussian kernel. This will be taken up in future work. However even if the accuracies are comparable, SVMs still suffer from the lack of interpretability which the rule learning framework



(a) Detection Rate (b) False Alarm Rate (c) Percentage of Correctly Classified Disks

Figure 2: Variation with number of rules

offers.

We partitioned the dataset according to their interface, i.e., Fibre channel (FC) and SATA. There were 6013 good disks and 2190 bad disks with a FC adapter and 19035 good disks and 1639 bad disks with a SATA adapter. The results are shown in Table I. For SATA disks, we could predict the failure of around 70% of them with only 1% FAR. For FC, we could predict the failure of around 73% of disks but only at 8% FAR. The reason for this could be that although the set of disk events was able to capture the conditions of failing hard disks, it was more suitable for prediction for disks using SATA than FC adapter.

On the disk dataset analyzed by Murray et al. [1], which consists of a total of 369 disks of which 178 are good and 191 are bad disks, the paper's best results using SVMs could only get around 60% detection at 3% FAR. On the contrary, using MLRules algorithm, the detection rate was around 96% at 3% FAR. However, we couldn't manage to bring the FAR below 3% which may be due to removal of incompatible attributes as discussed before.

D. *Interpretable Rules*

A sample rule generated by MLRules algorithm is shown in Figure 3. A collection of such rules form the rule ensemble. During classification, for each data instance, all the rules are progressively checked for satisfiability. If a particular rule is found to be satisfied, its weight is added to the class label it predicts. In the end, the class label with largest value is assigned to the given instance.

Rule-based techniques have a great advantage. The generated rules are easy to understand and are highly interpretable. For example, the rule shown in Figure 3 can be interpreted as *"If the given disk uses SATA interface, its size is greater than 285 GB, and the disk error and event counts are below stated thresholds then the disk can be said to be working fine with some probability"*. Such an interpretation is not possible in case of other techniques. The interpretation can also help in pruning insignificant rules from the classification model. Intuitively, a rule that is voting for class *good* should not contain *greater than* conditions for disk events, i.e. a disk is *good* if some event count exceeds a threshold. Such rules can be pruned from the final model.

Rule:
INTERFACE is SATA
MEDIA_ERR_TYPE1 $\leq$ 2.5
IO_REASSIGN_SUCCESS $\leq$ 77.0
MEDIA_ERR_TYPE2 $\leq$ 0.5
DISK_SIZE $\geq$ 285.0
IO_RECOVERED_ERROR $\leq$ 14.5
MEDIA_ERR_TYPE3 $\leq$ 133.0
REWRITE_DATA_FAILED $\leq$ 6.5

=> vote for class **GOOD**
with weight 0.08922562491818498

Figure 3: A sample rule generated by MLRules

Using disk events instead of just SMART attributes make the rules meaningful and increase their information content. The rules generated would be more intuitive and appealing considering the fact that they have conditions relating to disk event counts rather than, say, temperature or flyheight.

Rules can provide an insight on which disk events can be helpful in predicting failures. This could help in getting to the real cause of failures in hard disks which can help in improving the reliability of storage systems. We found that I/O recovered errors, events related to rewriting data, checksum errors, media errors, and transport errors were present in significant number of rules. This implies that such events have more predictive capabilities than others. Table II shows the list of events which had a sizable number of occurrences in rules.

Table I: Detection rate and FAR after partitioning data based on adapter

Adapter	Good Disks	Bad Disks	Detection Rate	FAR
SATA	19035	1639	70%	1%
FC	6013	2190	73%	8%

V. CONCLUSIONS AND FUTURE WORK

We have shown that the disk events have good value in predicting impending disk failure. Our results also indicate that by using disk events instead of just SMART attributes,

Table II: Number of occurrences of disk events in 100 rules

Event	Number of occurrences
Rewrite Data	71
I/O Recovered Error	63
Checksum Error	54
Transport Error	53
Media Error	49

significantly increases the prediction accuracy while keeping the false alarm rates to a minimum. We also tried to get an insight on disk events which could be associated with disk failures.

We have also shown that rule-based classifiers outperform the existing techniques for predicting impending hard disk failures. Generating rules is computationally inexpensive and less time-consuming. The generated rules are highly interpretable. These characteristics make rule-based learning techniques much more suitable for hard disk failure prediction. Using disk events, the rules generated are more meaningful and informative as compared to those generated using SMART attributes only.

In this paper, we used the cumulative error counts in hard disks for prediction of impending failures. Intuitively, hard disk failure seems to be of time-evolving nature. We plan to incorporate the time dimension also, which could improve the prediction accuracy and lower false alarms. Lot of research needs to be done on what attributes should be considered for prediction of hard disk failures.

REFERENCES

[1] J. F. Murray, G. F. Hughes, and K. Kreutz-Delgado, "Machine learning methods for predicting failures in hard drives: A multiple-instance application," *Journal of Machine Learning research*, vol. 6, 2005.

[2] G. F. Hughes, J. F. Murray, K. Kreutz-Delgado, and C. Elkan, "Improved disk-drive failure warnings," *IEEE Transactions on Reliability*, 2002.

[3] G. Hamerly and C. Elkan, "Bayesian approaches to failure prediction for disk drives," in *Proceedings of the Eighteenth International Conference on Machine Learning*, 2001.

[4] J. F. Murray, G. F. Hughes, and K. Kreutz-Delgado, "Hard drive failure prediction usig non-parametric statistical methods," in *Proceedings of ICANN/ICONIP*, 2003.

[5] E. Pinheiro, W. D. Weber, and L. A. Barroso, "Failure trends in a large disk drive population," in *Proceedings of the 5th USENIX Symposium on File and Storage Technologies (FAST 07)*, 2007.

[6] L. N. Bairavasundaram, G. R. Goodson, S. Pasupathy, and J. Schindler, "An analysis of latent sector errors in disk drives," in *Proceedings of the 2007 SIGMETRICS Conference on Measurement and Modeling of Computer Systems*, 2007.

[7] L. N. Bairavasundaram, G. R. Goodson, B. Schroeder, A. C. Arpaci-dusseau, and R. H. Arpaci-dusseau, "An analysis of data corruption in the storage stack," in *Proceedings of the 6th USENIX Symposium on File and Storage Technologies (FAST 08)*, 2008.

[8] K. Dembczynski, W. Kotlowski, and R. Slowinski, "Maximum likelihood rule ensembles," in *Proceedings of the 25th International Conference on Machine Learning*, 2008.

[9] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. volume 2, pp. 121–167, 1998.

[10] J. H. Friedman and B. E. Popescu, "Predictive learning via rule ensembles," *Annals of Applied Statistics*, vol. 2, p. 916, 2008.

[11] W. W. Cohen and Y. Singer, "A simple, fast, and effective rule learner," in *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, 1999, pp. 335–342.

[12] S. M. Weiss and N. Indurkhya, "Lightweight rule induction," in *Proceedings of the Seventeenth International Conference on Machine Learning*, 2000.

[13] W. Jiang, C. Hu, Y. Zhou, and A. Kanevsky, "Are disks the dominant contributor for storage failures? a comprehensive study of failure characteristics," in *Proceedings of the 6th USENIX Symposium on File and Storage Technologies (FAST 08)*, 2008.