Improving accuracy and adaptability of SSD failure prediction in hyper-scale data centers

Wenwen Hao
Samsung R&D Institute China
Xian, Samsung Electronics
Xian, China
wenwen.hao@samsung.com

Ben Niu
Tencent Holdings Limited
Shenzhen, China
benbniu@tencent.com

Yin Luo
Samsung R&D Institute China
Xian, Samsung Electronics
Xian, China
yin.luo@samsung.com

Kangkang Liu
Tencent Holdings Limited
Shenzhen, China
kirkliu@tencent.com

Na Liu
Samsung R&D Institute China
Xian, Samsung Electronics
Xian, China
na2.liu@samsung.com

ABSTRACT

The rapid expansion of flash-based solid state drives (SSDs) makes SSD failure an important factor impacting the reliability of storage systems in data centers. To improve the reliability and stability of storage systems, proactive failure prediction methods are investigated by researchers. Based on self-monitoring, analysis and reporting technology (SMART) logs, machine learning technologies are employed to improve the accuracy of SSD failure prediction. However, most of these works fail to achieve high true positive rate (TPR) or low false positive rate (FPR). Prior works also suffer from imbalance scale of healthy and failed data, weak predictability of SMART attributes as well as the variation issue of SMART distribution range. In order to improve TPR of SSD failure prediction, failure analysis and prediction methods are researched in this paper. First we conduct extensive failure analysis work on a dataset collected from Tencent datacenter. And then based on the analysis results, we propose a novel SSD failure prediction method. To address the main challenges of prior works, a new feature generation method, a random under-sampling based ensemble learning method (RUS_Ensemble) and a sorting strategy are proposed accordingly. The evaluation results on Tencent dataset show that our proposed method is able to improve TPR by 28%, while bringing a relative low FPR which is less than 1%.

Keywords

SSD, failure prediction, SMART, Machine Learning

1. INTRODUCTION

In this cloud computing and big data era, the reliability of a cloud storage system relies on the storage devices it builds on. Flash-based solid state drives (SSDs) as a high-performance alternative to hard disk drives (HDDs) have been widely used into storage systems. Both the number and data capacity of SSDs grow steadily every year. Besides, media-level technology is also employed by SSD vendors to increase the storage density of SSDs. However, increasing in storage density may bring about decreasing in endurance, retention, and reliability of SSDs. Therefore, SSDs have become one of the main sources of failures in datacenters nowadays. A hyper-scale datacenter with millions of SSDs faces multiple SSD failures every day, which has a great impact on reliability of storage systems.

Faced with SSD failures, reactive fault tolerance and proactive failure prediction methods are adopted to improve the reliability of storage systems. Reactive fault tolerance like erasure codes and data redundancy mechanisms aims to help storage application recover from SSD failures. However, the detection of SSD failures may be behind their occurrence if only the reactive fault tolerance strategy is employed. Therefore, proactive failure prediction methods are investigated to improve the reliability and stability of storage systems. These methods involve predicting an SSD failure before it actually happens, and therefore, a disk replacement advice can thus be informed to the datacenter.

Proactive failure prediction is generally based on SMART attributes. In recent years, machine learning method is adopted to build binary classification model on a disk drives to make proactive failure prediction, and the model is always built on a large-scale dataset consisting of SMART logs. By capturing certain prediction rules from large-scale dataset, the model is able to make proactive failure prediction by taking a single snapshot of the SMART attributes as the input. Various machine learning methods have been proposed to improve the accuracy of SSD failure prediction.

Prior study in the field of predicting storage device failure primarily focuses on traditional HDDs. SMART logs are originally designed to detect and report various indicators of drives' reliability, and therefore, naive threshold-based method is used for failure prediction. Furthermore, machine learning methods like Bayesian [8], rank-sum [17] and multi-instance Naive Bayes (mi-NB) [18] are employed to improve accuracy of disk failure prediction. But all of these methods are evaluated on small datasets in their corresponding work. Anomaly detection methods [24, 22] are also investigated and adopted in disk failure prediction but they have not improved the situation yet. Recently, tree-based methods [10,

Workshop on AI in Networks and Distributed Systems (WAIN) 2021 Milan,Italy

4] are proved to perform well on prediction accuracy and interpretability. The aforementioned methods do not take time sequence information into consideration in disk failure prediction. Most recent studies [25, 7, 11, 13] focus on modeling time sequence dependence and are proved successful in proactive failure prediction.

Prior techniques and findings for HDDs are not applicable to SSDs due to fundamental difference in architecture [6]. Research that particularly focusing on SSDs [3, 5, 14, 26] is limited to specific errors in controlled environment. Several studies [16, 21, 9] also analyze the effects of correlated factors on SSD reliability, but they are aimed at guiding the design of redundancy protection for high storage reliability. Machine learning method is also adopted by some works [2, 19] for making proactive failure prediction. But most of these works fail to achieve high TPR or low FPR. Additional device-level attributes are used by Farzaneh [15] and Chandranil [6] to improve the performance of SSD failure prediction. However, the method is limited to certain type of SSDs, since the attributes used are not accessible for all SSDs.

Prior works face the following three major challenges. First, the scale of healthy SMART observations is many times larger than that of failed SMART observations, thus posing data imbalance problem to machine learning methods. Second, static value of SMART attributes can hardly indicate SSD failures. Third, the distribution range of SMART attributes varies every day due to different workloads and application types, and therefore, there is a bias in the output of failure prediction model. Faced with these three issues, prior works can only achieve low TPR value. A novel SSD failure prediction scheme is proposed in this paper to help improve TPR and bring a relative low FPR value of less than 1%.

We present an in-depth data-driven analysis on SSD failure and propose a novel SSD failure prediction method based on the analysis result. The main contributions of this paper are as follows:

- Based on a dataset collected from Tencent datacenter, we conducted extensive analysis on SSD failure. By categorizing SSD failures into several types, we observe that there is a distinct failure pattern for each type.

- Based on the analysis result, we propose an SSD failure prediction solution to improve the performance and solve the issues in prior works.

- To solve the problem of weak correlation between SMART attributes and SSD failure, new features are generated in data pre-processing stage.

- To solve the data imbalance and distribution variation problem, RUS_Ensemble model and sorting strategy are proposed in the SMART failure prediction stage.

- A window-based SSD failure prediction is developed based on the failure prediction results of SMART observations to decrease FPR.

2. SSD FAILURE ANALYSIS

As mentioned above, the existing SSD failure prediction solutions fall short in either the performance or generality. Three major challenges listed above pose significant bottlenecks to the study of SSD failure prediction. In this section, extensive analysis is conducted on Tencent dataset for a better understanding of SSD failure characteristics, which can in turn help to optimize our SSD failure prediction solution. Specifically, time series dependency of SMART observations is analyzed in this section.

The correlation analysis between static value of SMART attributes and SSD failures is helpful to the design of the failure prediction scheme, so we conduct the correlation analysis on Tencent dataset.In our correlation analysis, the absolute value of Pearson coefficient is used to measure the correlation between SMART attributes and SSD failures, and a value close to zero implies weak predictability. Figure 1 shows the correlation coefficient of the top 10 most indicative SMART attributes. It reveals a weak correlation between SMART attributes and SSD failures.

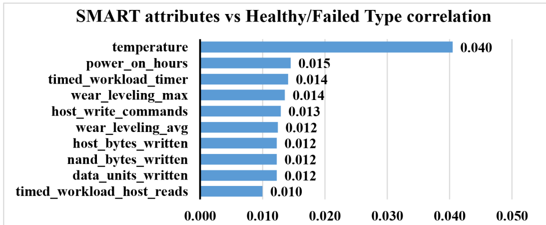

Figure 1: Correlation analysis of SMART attributes

As the static value of SMART attributes is weakly correlated to SSD failures, time series analysis is conducted to capture the difference in changing trend of healthy and failed drives. SSD failure may be caused by a variety of factors and there is also difference in failure symptoms on SMART attributes. In most cases, there is no idea about the root causes when SSD failure happens, but we can categorize SSD failures into several types according to their impact on application layer. In this section we focus on major types of SSD failures and conduct failure analysis on each type.

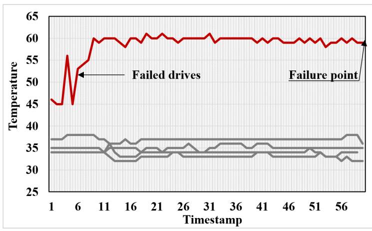

Figure 2: Temperature trend of healthy and dropping-out SSDs

Dropping out: Failure of dropping out accounts for a large portion of failed data in Tencent dataset. To test whether the SMART attributes exhibit distinct patterns between healthy SSDs and dropping-out SSDs, comparative changing trend of healthy and dropping-out SSDs is plotted. There is a changing point on time sequence of temperature or WAF several days before dropping-out failure happens. Figure 2 shows that the temperature of SSDs with dropping-out failure increases sharply before actual failure happens, while the temperature of healthy drives keeps stable between 35°C to 40°C. Another failure pattern in Figure 3 shows that as the data units written to or read from SSDs increase, the WAF value of dropping-out SSDs increases sharply from a normal value to a relatively large one before actual failure happens, while the WAF value of healthy SSDs on the same sever is always keep stable at a normal value. We also notice that most of SSDs with dropping-out failure exhibit at least one failure pattern as shown in Figure 2 and Figure 3.

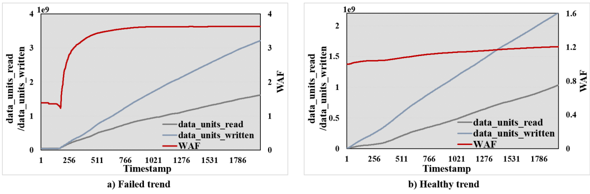

Figure 3: WAF trend of healthy and dropping-out SSDs

Media error: A large number of failed SSDs in Tencent dataset are of media error type. Media error is caused by uncorrectable error during a read operation and in SMART logs there is an indicator attribute (media_error) specifically for monitoring this kind of failure. As shown in Figure 4, when the data read from or written into SSDs increases, the value of indicator attribute of failed SSDs with media error increases accordingly, while that of healthy drives is always 0. We find that for half of the media error-related failures, the value of indicator attribute increases sharply from 0 to a threshold at the time failure happens; for the other half, the value increases gradually several days before actual failure. The indicator attribute shows distinct failure pattern for media error-related failure, but in some cases this kind of failure pattern lags behind actual failure time.

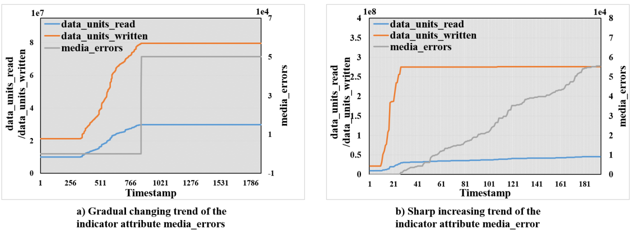

Figure 4: Trend of workload and indicator attribute for media error failure

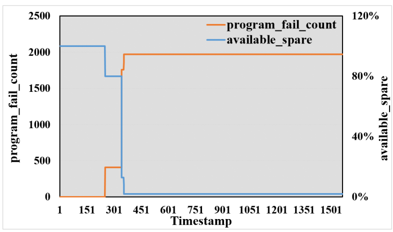

Figure 5: Failure pattern of bad blocks-related failure

Bad blocks: Bad block-related failure is another major failure in Tencent dataset. As shown in Figure 5, compared with healthy SSDs there is a changing point on SMART attributes of program_fail_count and available_sapre about several weeks before bad block-related failure happens. The two attributes of healthy SSDs always keep at a constant value, while for bad block-related failed SSDs, there will be an increasing trend on program_fail_count and deceasing trend on available_spare. Bad block-related failure happens when the value of available_spare decreases to a threshold. The analysis result reveals that this type of failure can be detected by capturing changing trend of SMART attributes of program_fail_count and available_spare.

3. THE PROPOSED METHOD

Taking the main challenges mentioned previously into consideration, an SSD failure prediction scheme is proposed in this paper and detailed description of the scheme will be presented in this section. As shown in Figure 6, the SSD failure prediction scheme consists of feature processor, SMART failure predictor and SSD failure predictor. The SMART logs collected from client are first sent to feature processor to construct indicative input for ML model. The output of feature processor is then fed into SMART failure predictor to predict failure on SMART observations. Finally, SSD failure predictor is followed to make final prediction based on the output of SMART failure predictor.

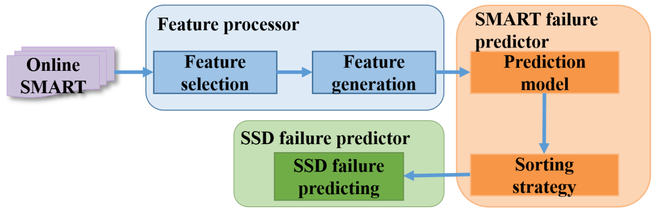

Figure 6: Block diagram of our proposed failure prediction scheme

3.1 Feature processor

As mentioned in Section 1, weak predictability of SMART attributes is one of the main challenges in SSD failure prediction. Based on the time series analysis in Section 2, feature processor is not only responsible for selecting indicative features from SMART attribute but also generating new features to capture time series information, thus solving the weak predictability problem. The feature processor first selects the most important attributes from SMART logs as the raw features, and then generates new features to capture time series information. Both the selected raw features and the generated new features are output to SMART failure predictor to make failure prediction.

Feature selection is performed at the first step of the feature processor, since not all of the SMART attributes are correlated to SSD failure. Indicative features selected in this process (see Table 1) is based on the failure analysis result. Raw feature power_on_ours in the table is selected as the time series indicator and the other 8 features are selected since they are indicative to at least one type of failure mentioned in Section 2.

To capture time series-related information, feature generation is performed after the feature selection process. According to the failure analysis result, differential and WAF features are generated to improve the predictability of the input of ML model. As shown in Table 1, differential value of two types of attributes including workload-related attributes and temperature is generated to capture time series-related features. WAF is also generated in this process by calculating the ratio of nand_bytes_written to the host_bytes_written.

Table 1: Selected and generated features of feature processor

Selected raw features	Generated new features
data_units_read	data_units_read_diff
data_units_written	data_units_written_diff
nand_bytes_written	WAF
host_bytes_written	
Temperature	temperature_diff
power_on_hours	-
media_errors	-
available_spare	-
program_fail_count	-

3.2 SMART failure predictor

SMART failure predictor is responsible for making failure prediction on SMART observations. It focuses on the data imbalance and distribution the variation issue of SMART attributes. RUS_Ensemble and sorting strategy are proposed in this stage to improve the accuracy of failure prediction on SMART attributes, thus ensuring high accuracy of SSD failure prediction.

RUS_Ensemble method: The base model selected is crucial when machine learning technology is employed to predict SSD failures. In our case, SMART attributes are weakly correlated to SSD failures and distinct failure pattern can be captured from time series analysis. Therefore, LSTM model is selected in this paper to model time series-related dependency of SMART attributes.

With LSTM as the base model, RUS_Ensemble method is proposed in this paper to solve the data imbalance problem. Random under-sampling (RUS) is commonly used to solve data imbalance by abandoning part of majority-class data. However, the drawback lies in the loss of data information caused in RUS process. By combining RUS strategy and ensemble learning, RUS_Ensemble is proposed to avoid the loss of data information when dealing with imbalance problem. The main idea of RUS_Ensemble is to train each of the n base models with a subset obtained from whole training set using under-sampling strategy. Since the training set of base model consists of 1/n of the majority-class data and the whole minority-class data and the majority-class data selected by n base models is exclusive, RUS_Ensemble is able to avoid data information loss when solving imbalance problem (see Table 2).

Table 2: RUS_Ensemble vs RUS/Ensemble model

	RUS	Ensemble	RUS_Ensemble
Data Information Loss	Yes	No	No
Data Imbalance Problem	No	Yes	No

Figure 7 shows an overview of the training and predicting phase of RUS_Ensemble method. In the training stage, ensemble learning is employed to train n base models for failure prediction. For each base model, RUS strategy is adopted to construct a training dataset containing 1/n of majority-class data and whole minority-class data. In addition, majority-class data is sampled without replacement to ensure that 1/n of majority-class data selected by each base model is exclusive. In the predicting phase, each base model outputs a risk score of the SMART observation and a final risk score obtained by using averaging strategy is output to the sorter to predict failure.

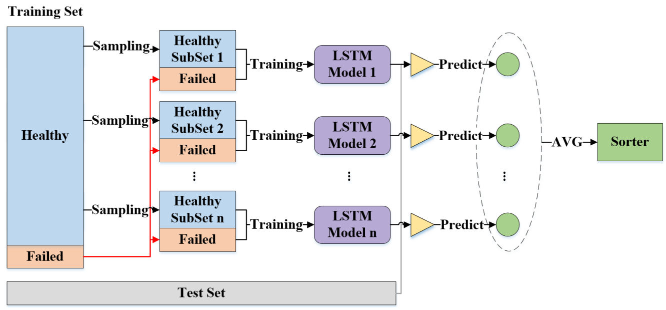

Figure 7: Overview of RUS_Ensemble method

Sorting strategy: Failure prediction of SMART observations is a typical binary classification issue in machine learning. Conventionally, when using a binary classification model to predict failure, both the risk score and healthy/failed type is output by prediction model. In most cases, if the risk score of a SMART observation it is greater than a certain threshold (commonly 0.5), it is predicted as failed; otherwise, predicted as healthy. However, the distribution variation of SMART attributes problem results from changing in workload and application type, brings about fluctuation in risk score. Therefore, predicting failure based on a single threshold for all SMART observation may lead to performance degradation. A higher threshold may result in the decrease of TPR while a lower one leads to the increase of FPR.

For a stable storage system we can assume that daily failure ratio of SSDs and SMART observation is stable. Also, there is no obvious fluctuation in the distribution of SMART attributes within a day, as a result, the risk score of almost all failed observations is greater than that of healthy observations. Based on these two assumptions, sorting strategy is proposed to solve distribution variation of SMART attributions. As a post-process strategy, the first step is to categorize the SMART observations into several buckets according to the collection date. For each collection date, we sort the SMART observations in descending order according to the risk score obtained from RUS_Ensemble model, and give failure prediction for the top p percent (a tunable parameter) of observations and health prediction for the remaining.

3.3 SSD failure predictor

In previous work, an SSD is predicted as failed if the SMART predictor makes failure prediction for its SMART observation collected at any time. It would be arbitrary to trigger an SSD failure alarm based on failure prediction to each SMART observation, which may lead to the increase of FPR. Based on this, an SSD failure predictor is introduced in this paper to predict SSD failure using sliding time window. The workflow of SSD failure predictor is shown in Figure 8. For each SSD, SMART logs are collected periodically and at each collection point in time axis a failure/health prediction is obtained from SMART failure predictor. Then a time window sliding on time axis is used to predict SSD failure based on the failure/health prediction of SMART logs. If the number of failed SMART observations in the time window is greater than a certain threshold, a failure prediction of an SSD is triggered.

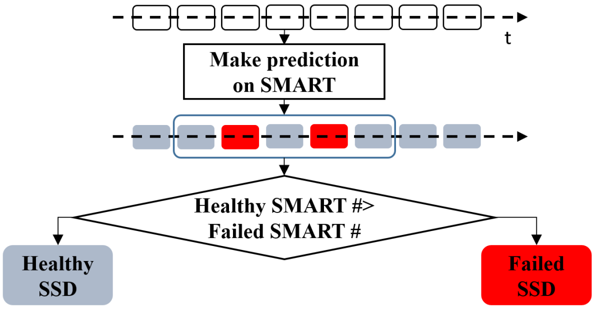

Figure 8: Workflow of SSD failure predictor

4. RESULTS AND ANALYSIS

To evaluate the performance of our proposed SSD failure prediction method, extensive experiments are conducted on Tencent dataset. Our dataset contains SMART data from over 100,000 drives over a period of about one year. About 60M observations with 40 different SMART attributes are contained in the dataset. Among the 60M observations, about 58,702 data samples collected from 114 drives are labeled as failed.

In this section, TPR, FPR and AUC score are used to measure the effectiveness of our approach. TPR represents the proportion of TP among all actually failed drives. FPR indicates the ratio of FN among all the healthy drives. Since the target data of our failure prediction model is highly imbalanced, AUC score is also used in this paper to evaluate the performance of different approaches.

To have a comprehensive analysis and evaluation of our proposed method, extensive tests are conducted. First, we compare our proposed method with four baselines used in [22]. And then, the lead time of failure prediction point with respect to actual failure occurrence point is evaluated to measure the sensitivity of our proposed method.

4.1 Accurate prediction of SSD failures

Our proposed method is compared against Bayes classifier (Bayes) [20], random forest (RF) [12], gradient boosted decision tree (GBDT) [23] and LSTM [1], and Table 3 shows the comparison result. The main drawback of prior works is that they can only achieve very low TPR value. As the table shows, LSTM and GBDT can only achieve a TPR less than 10%. Bayes classifier is able to improve TPR to 23% but it brings about a very high FPR value which is close to 10%. Compared with prior works, our proposed RUS_Ensemble method improves TPR value by about 28%, while bringing a relative low FPR value of less than 1%.

In a datacenter, false positive prediction can also cause maintenance overhead to datacenters. Limiting FPR into a relative low level is very important for SSD failure prediction. Except for the Bayes, both our method and prior works are able to achieve very low FPR which is less than 1%, which could reduce maintenance cost caused by false positive prediction. With a low FPR, our method can achieve

Table 3: Comparison of our method and 4 prior works

	TPR(%)	FPR(%)	AUC
Bayes	23.810	9.811	0.473
RF	9.523	0.00403	0.584
GBDT	4.762	0.00134	0.439
LSTM	9.524	0.00941	0.751
RUS_Ensemble	38.095	0.758	0.755

higher TPR which is 38%. Therefore, the main contribution of our method is to reduce the cost of SSD failure for datacenters, only bringing about very low overhead for false positive prediction.

4.2 Prediction time analysis

To verify whether our proposed method can predict SSD failure ahead of or behind the actual failure, we also conduct an evaluation on prediction time and actual occurrence time of SSD failure. In the evaluation, we categorize the true positive drives into three groups according to whether the failure prediction time is ahead of, behind or within the actual failure day. Figure 9 shows the ratio of true positive drives of each group. It reveals that only 25% of the true positive drives is predicated several days ahead of actual failure, while for most true positive drives, failure prediction time is either behind or within the actual failure day. Our further analysis on Tencent dataset explains this result. The reason may be that a change point on SMART attributes of failed drives always occurs only a few hours before or several days after actual failure happens. Therefore, it is difficult to make failure prediction before actual failure time by only using SMART attributes. More research should be done on introducing more internal attributes (like Telemetry features) to enable early prediction of SSD failure.

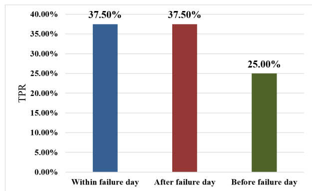

Figure 9: Ratio of true positive drives in different prediction period

5. CONCLUSIONS

In this paper, we provide a comprehensive study of SSD failure analysis and prediction. Based on a dataset collected from 100,000 drives over a period of about one year, time series-related feature of SSD failure is analyzed in this paper. And then based on the failure analysis result, we propose a novel SSD failure prediction method. In our method, new features are generated to capture indicative failure pattern of SSDs and RUS_Ensemble model is proposed to solve data imbalance. Moreover, a sorting strategy is proposed for post-processing stage to solve distribution variation problem. Furthermore, extensive test and analysis is conducted to evaluate the performance of our proposed method. The result shows that our proposed method is able to improve TPR value by 28%, while bringing a relative low FPR value of less than 1%.

6. REFERENCES

[1] Long short-term memory. Neural Computation, 9(8):1735–1780, 1997.

[2] J. Alter, J. Xue, A. Dimnaku, and E. Smirni. SSD failures in the field: symptoms, causes, and prediction models. In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC 2019, Denver, Colorado, USA, November 17-19, 2019, pages 75:1–75:14. ACM, 2019.

[3] H. Belgal, N. Righos, I. Kalastirsky, J. Peterson, R. Shiner, and N. Mielke. A new reliability model for post-cycling charge retention of flash memories. In 2002 IEEE International Reliability Physics Symposium. Proceedings. 40th Annual (Cat. No.02CH37320), pages 7–20, 2002.

[4] M. M. Botezatu, I. Giurgiu, J. Bogojeska, and D. Wiesmann. Predicting disk replacement towards reliable data centers. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016, pages 39–48. ACM, 2016.

[5] Y. Cai, Y. Luo, E. F. Haratsch, K. Mai, and O. Mutlu. Data retention in MLC NAND flash memory: Characterization, optimization, and recovery. In 21st IEEE International Symposium on High Performance Computer Architecture, HPCA 2015, Burlingame, CA, USA, February 7-11, 2015, pages 551–563. IEEE Computer Society, 2015.

[6] C. Chakrabortti and H. Litz. Improving the accuracy, adaptability, and interpretability of SSD failure prediction models. In SoCC '20: ACM Symposium on Cloud Computing, Virtual Event, USA, October 19-21, 2020, pages 120–133. ACM, 2020.

[7] Chang, Xu, Gang, Wang, Xiaoguang, Liu, Dongdong, Guo, Tie-Yan, and Liu. Health status assessment and failure prediction for hard drives with recurrent neural networks. IEEE Transactions on Computers, 2016.

[8] G. Hamerly and C. Elkan. Bayesian approaches to failure prediction for disk drives. 2003.

[9] D. Jauk, D. Yang, and M. Schulz. Predicting faults in high performance computing systems: an in-depth survey of the state-of-the-practice. In Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC 2019, Denver, Colorado, USA, November 17-19, 2019, pages 30:1–30:13. ACM, 2019.

[10] L. Jing, X. Ji, Y. Jia, B. Zhu, and X. Liu. Hard drive failure prediction using classification and regression trees. In 2014 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, 2014.

[11] A. Lh, A. Lh, B. Zx, C. Tj, and A. Hq. A disk failure prediction method based on lstm network due to its individual specificity. Procedia Computer Science, 176:791–799, 2020.

[12] A. Liaw and M. Wiener. Classification and regression by randomforest. R News, 23(23), 2002.

[13] S. Lu, B. Luo, T. Patel, Y. Yao, D. Tiwari, and W. Shi. Making disk failure predictions smarter! In 18th USENIX Conference on File and Storage Technologies, FAST 2020, Santa Clara, CA, USA, February 24-27, 2020, pages 151–167. USENIX Association, 2020.

[14] Y. Luo, S. Ghose, Y. Cai, E. F. Haratsch, and O. Mutlu. Improving 3d NAND flash memory lifetime by tolerating early retention loss and process variation. Proc. ACM Meas. Anal. Comput. Syst., 2(3):37:1–37:48, 2018.

[15] F. Mahdisoltani, I. Stefanovici, and B. Schroeder. Improving storage system reliability with proactive error prediction. In 2017 Usenix Annual Technical Conference (ATC '17), July 2017.

[16] J. Meza, W. Qiang, S. Kumar, and O. Mutlu. A large-scale study of flash memory failures in the field. Acm Sigmetrics Performance Evaluation Review, 43(1):177–190, 2015.

[17] J. F. Murray, G. F. Hughes, and K. Kreutz-Delgado. Hard drive failure prediction using non-parametric statistical methods. 2003.

[18] 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, 6(27):783–816, 2005.

[19] I. Narayanan, D. Wang, M. Jeon, B. Sharma, L. Caulfield, A. Sivasubramaniam, B. Cutler, J. Liu, B. M. Khessib, and K. Vaid. SSD failures in datacenters: What, when and why? In Proceedings of the 2016 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Science, Antibes Juan-Les-Pins, France, June 14-18, 2016, pages 407–408. ACM, 2016.

[20] I. Rish. An empirical study of the nave bayes classifier. journal of universal computer science, 2001.

[21] B. Schroeder, R. Lagisetty, and A. Merchant. Flash reliability in production: The expected and the unexpected. In 14th USENIX Conference on File and Storage Technologies, FAST 2016, Santa Clara, CA, USA, February 22-25, 2016, pages 67–80. USENIX Association, 2016.

[22] Y. Wang, Q. Miao, T. Duan, K. L. Tsui, and M. G. Pecht. Online anomaly detection for hard disk drives based on mahalanobis distance. IEEE Transactions on Reliability, 62(1):136–145, 2013.

[23] J. Ye, J. H. Chow, C. Jiang, and Z. Zheng. Stochastic gradient boosting distributed decision trees.

[24] W. Yu, M. Qiang, and M. Pecht. Health monitoring of hard disk drive based on mahalanobis distance. IEEE, 2011.

[25] Y. Zhao, X. Liu, S. Gan, and W. Zheng. Predicting disk failures with HMM- and hsmm-based approaches. In Advances in Data Mining. Applications and Theoretical Aspects, 10th Industrial Conference, ICDM 2010, Berlin, Germany, July 12-14, 2010. Proceedings, volume 6171, pages 390–404. Springer, 2010.

[26] M. Zheng, J. Tucek, F. Qin, and M. Lillibridge. Understanding the robustness of SSDS under power fault. In K. A. Smith and Y. Zhou, editors, Proceedings of the 11th USENIX conference on File and Storage Technologies, FAST 2013, San Jose, CA, USA, February 12-15, 2013, pages 271–284. USENIX, 2013.