applied sciences

MDPI

Article

Feature Selection for Improving Failure Detection in Hard Disk Drives Using a Genetic Algorithm and Significance Scores

Wasim Ahmad [1,†], Sheraz Ali Khan [1,†], Cheol Hong Kim [2,†] and Jong-Myon Kim [1,*,†]

[1] School of Electrical, Electronics and Computer Engineering, University of Ulsan, Ulsan 44610, Korea; wasimahmad.qc@gmail.com (W.A.); sherazalik@gmail.com (S.A.K.)

[2] Department of Computer Engineering, Chonnam National University, Gwangju 61186, Korea; chkim22@chonnam.ac.kr

* Correspondence: jmkim07@ulsan.ac.kr; Tel.: +82-52-259-2217

† These authors equally contributed to this manuscript.

Received: 17 March 2020; Accepted: 28 April 2020; Published: 4 May 2020

Abstract: Hard disk drives (HDD) are used for data storage in personal computing platforms as well as commercial datacenters. An abrupt failure of these devices may result in an irreversible loss of critical data. Most HDD use self-monitoring, analysis, and reporting technology (SMART), and record different performance parameters to assess their own health. However, not all SMART attributes are effective at detecting a failing HDD. In this paper, a two-tier approach is presented to select the most effective precursors for a failing HDD. In the first tier, a genetic algorithm (GA) is used to select a subset of SMART attributes that lead to easily distinguishable and well clustered feature vectors in the selected subset. The GA finds the optimal feature subset by evaluating only combinations of SMART attributes, while ignoring their individual fitness. A second tier is proposed to filter the features selected using the GA by evaluating each feature independently, using a significance score that measures the statistical contribution of a feature towards disk failures. The resultant subset of selected SMART attributes is used to train a generative classifier, the naïve Bayes classifier. The proposed method is tested on a SMART dataset from a commercial datacenter, and the results are compared with state-of-the-art methods, indicating that the proposed method has a better failure detection rate and a reasonable false alarm rate. It uses fewer SMART attributes, which reduces the required training time for the classifier and does not require tuning any parameters or thresholds.

Keywords: failure detection; hard disk drives; genetic algorithms; classification; feature selection

1. Introduction

The annual shipments of tablet computing devices surpassed personal computers in 2013, making flash memory the predominant mode of data storage for digital consumers [1]. However, the size of the digital universe is expected to be around 40 trillion gigabytes, which is roughly 5200 gigabytes of data for every living person on earth [2]. For a long time to come, consumers of digital data would rely on relatively cheaper storage options for backup and archival purposes, i.e., local or cloud storage afforded by magnetic hard disk drives (HDD) [1]. HDD are generally regarded as reliable components as their mean time to failure (MTTF) typically lies between 1 million to 1.5 million hours, as per manufacturer specifications. This would suggest a maximum annual failure rate (AFR) of 0.88%; however, field observations indicate that annual disk replacement rates (ARR) of 2–4% are common and, in some systems, replacement rates of up to 13% have also been reported [3–5]. Moreover, HDD are the most frequently replaced hardware components in large-scale information technology (IT) infrastructure [6], and, in some large datacenters, 70–80% of the known failures have been caused by failing HDD [7,8]. Thus, detecting a failing HDD is important to ensure data availability and to avoid permanent data loss.

Hard disk drives are complex electromechanical devices and their failure can be caused by a number of factors, which may not always be easy to determine. A review of the available literature reveals that three approaches have been considered to determine the characteristics of failed HDD and then to use that knowledge to predict an impending failure. The first approach explores the use of external sensors, the second approach analyzes logs and replacement records of failed drives, whereas techniques based on the third approach utilize self-monitoring, analysis, and reporting technology (SMART) attributes that are internally recorded in all HDD. In [9], the use of accelerometers and acoustic emissions (AE) sensors has been investigated. However, the authors conclude that the degradation signatures obtained from the vibration acceleration and AE signals are weak, inconsistent, and do not follow a trend that is indicative of the degradation of the HDD. Moreover, the use of accelerometers and AE sensors is prohibitive in terms of the associated costs, given the number of sensors required to monitor large-scale IT infrastructure. Hence, they propose using three SMART attributes instead, to evaluate the condition of the HDD. In [3,4,6,10], the authors analyze logs and replacement records of more than 100, 000 hard disks that were determined to be faulty and hence replaced by the customers. However, these studies do not investigate the causes of disk failures; rather, they provide an analysis of the data from a reliability perspective, and hence do not offer any useful clues for predicting a failing HDD. Nevertheless, these studies do offer important conclusions. First, HDD failure rates are influenced by factors such as workloads, "duty cycles", "powered-on-hours", and the temperature and humidity of the environment they operate in. Second, the actual replacement rates for HDD are much higher than those suggested by manufacturer specifications, roughly by a factor of 2 to 10 for drives less than five years old, and by a factor of 30 for older drives. Third, the replacement rates for serial advanced technology attachment (SATA), small computer system interface (SCSI), and fiber channel (FC) drives are more or less the same. The majority of the techniques [7,11–24] developed to detect failing HDD use the third approach, i.e., utilizing SMART attributes to detect impending disk failure. SMART was originally named drive failure prediction (DFP) and was adopted by the HDD industry in 1995 [11,13]. It monitors different operational parameters and features of a disk drive, such as *temperature*, *read-write errors*, and *power-on hours*, which can be used by the basic input output system (BIOS) or client software to detect an imminent failure. The simplest way to use these attributes to detect a failing disk has been to determine whether a particular SMART attribute has exceeded a predetermined threshold or not [11,13]. However, among the various SMART attributes, identifying the ones that are more indicative of a failing disk drive is not easy. There is no broad consensus on the precursors of hard disk failure among researchers. For example, the study in [16] concludes that there is no significant correlation between disk failure and its temperature and activity level; this assertion, however, is disputed by later studies [7,22]. In [7], the authors find a direct relationship between increasing AFRs and increasing disk temperatures, whereas, in [22], hard disk failures are found to be inversely related to the temperature, i.e., hotter disks are less likely to fail than cooler disks. The authors assume that this may be due to higher relative humidity experienced by the cooler disks. It's important to note that both [7,22] studies are based on the IT infrastructures of the same company. In [16], the authors acknowledge a high correlation between several SMART attributes and disk failures; they, however, rule out their utility at predicting the failure of individual disk drives. Nevertheless, several studies have demonstrated that SMART features can, in fact, be used to predict impending failure in disk drives [9,12,13,15,17–21,23]. Many of these studies [9,12,19,21,23,24] do not provide a sound mechanism for selecting the SMART attributes that can serve as the most effective precursors for an impending failure. The selection of good SMART attributes is very important for improving the detection accuracy and reducing the false alarm rate (FAR). Moreover, because of the inherent class imbalance in the available data, i.e., an average ARR of 4% means that 96% of drives are healthy and only 4% fail, reducing the FAR is even more significant, e.g., for an infrastructure with 100, 000 disk drives, a reduction of 0.01% in the FAR would result in 1000 fewer replacements of otherwise healthy disk drives. Thus, even a small improvement in the FAR would save the time, effort, and bandwidth required to perform unwarranted disk replacements and the associated data transfers.

In this paper, a two-tier approach is proposed to select the most indicative SMART attributes as precursors to a failing hard disk. In the first tier, a genetic algorithm (GA) is used for feature (or attribute) subset selection, whereas, in the second tier, a feature significance function is used to examine the selected subset of features and find those SMART attributes that would result in the best detection accuracy while minimizing the FAR. The GA selects the optimal SMART attributes using a mechanism that is analogous to biological evolution, where the principles of natural selection prune the sub-optimal characteristics in living organisms, over successive generations. In this study, the GA uses the ratio of the compactness of a class to the separability between classes as the objective function, and finds the optimal subset of SMART attributes that minimizes the objective function over successive generations. While the GA evaluates the SMART attributes in different combinations, the feature significance function considers individual attributes and discards those that are not significantly related to drive failures. A naïve Bayes (NB) classifier is then used to construct a model for the selected SMART attributes, which can differentiate healthy drives from failure-prone drives. Moreover, because of the inherent imbalance between the number of failed and healthy drives, random under-sampling of the majority class is used to mitigate the effects of class imbalance on the performance of the classifier.

The remainder of this paper is organized as follows. Section 2 presents the details of the proposed methodology, while Section 3 describes the SMART attributes dataset used to test the proposed methodology. Section 4 provides a discussion of the results obtained, and Section 5 presents the conclusions of this work.

2. The Proposed Methodology

The proposed methodology for detecting failing hard disks is illustrated in Figure 1. It utilizes SMART attributes to determine whether a disk is healthy or failing. SMART attributes are collected by modern HDD for self-monitoring purposes. These attributes are viewed as feature vectors in a high-dimensional space; the dimensionality of the feature space is determined by the number of selected SMART attributes. It is hypothesized that, for healthy and failing drives, these feature vectors will tend to form different clusters, which would be distinguishable in the high-dimensional feature space. A classifier could then be trained to differentiate between these two clusters, and hence determine whether a disk is failing or not using SMART attributes. Nevertheless, not all SMART attributes would be equally helpful in leading to easily separable clusters in the feature space; this would adversely affect the classifier’s performance [25,26]. Hence, feature selection is used to select the optimal subset of features that would result in more compact and clearly separable clusters of vectors of the selected features. In addition to improving the predictive performance of the classifier, feature selection reduces the measurement and storage requirements, as well as the training and prediction times [27]. This study proposes a two-tier approach for feature selection using a GA and feature significance function.

While the GA evaluates different combinations of features to select the optimum subset, the feature significance function is used to individually evaluate each member of this optimal feature subset using significance scores. It discards those features that are determined to be statistically insignificant in contributing towards disk failure. The final set of selected features is then used to train a Bayes classifier that can detect a failing HDD.

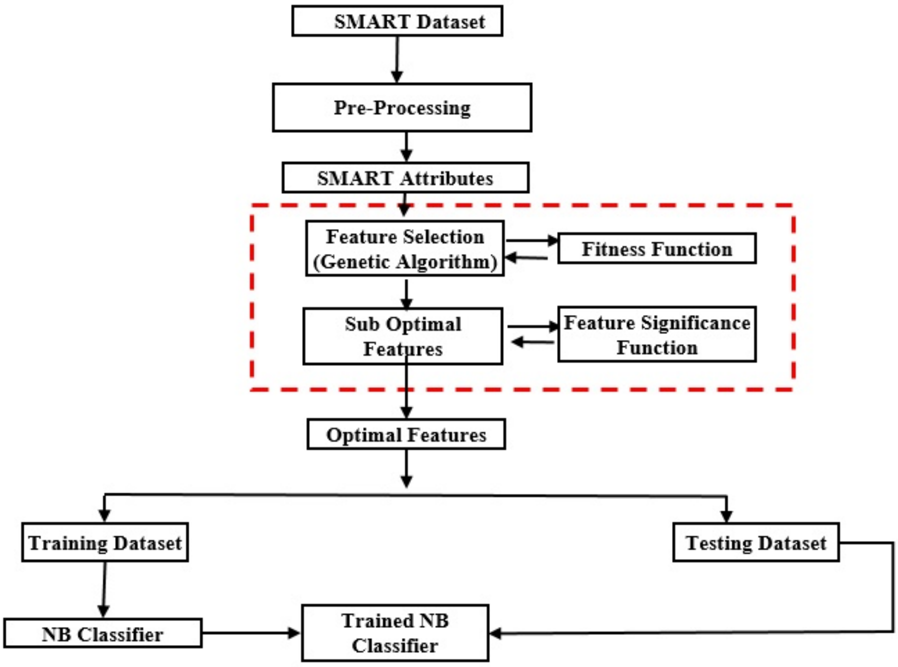

Figure 1. The proposed methodology for detecting failing disk drives using a genetic algorithm and a feature significance function.

2.1. Feature Selection

Machine learning techniques construct models of labeled objects in order to distinguish them from each other. These objects are described by feature vectors. The accuracy of these models depends upon the quality of these features, i.e., the more discriminative the features, the more accurate the machine learning model. However, not all the features used to describe a given object may be useful in distinguishing it from other objects. That is, some features may be irrelevant or redundant and hence may adversely affect both the classification accuracy of the machine learning model and the time required to construct it [25,28,29]. Therefore, feature selection algorithms are an essential part of many machine learning applications as these algorithms help in removing the irrelevant and redundant features to improve the classification accuracy, training time required to construct the model and the reduce the required number of training for better generalization [28–33].

Feature selection methods can be broadly divided into the following three categories [28]:

1. **Filter Based Methods:** These methods use some sort of fitness function to first evaluate and rank different features, and then select a subset of features that have fitness function values above a certain threshold. In essence, these methods filter out the bad features first and then construct the machine learning model. This approach is usually more efficient [28,29], but its performance depends upon the quality of the fitness function. The feature selection method used in this study can be categorized as a filter based method.
2. **Wrapper Based Methods:** These methods do not filter out the bad features before constructing the machine learning model. Rather, they use the classifier to filter out the bad features. For example, different combinations of features may be used by the classifier, and the combination of features that yields the highest classification accuracy may be selected as the best set of features. This approach can be very time consuming and may only result in a sub-optimal solution [28,29].
3. **Embedded or Hybrid Methods:** As the name suggests, these methods use an embedded or a hybrid approach. Unlike wrapper based methods, which iterate through different combinations of the features and may select the best subset of features on the basis of the accuracy of the classifier, these methods do not involve such iterative use of the classifier, which improves their speed.

Similarly, unlike the filter based approaches, these methods do not use a separate fitness function to rank different features. Rather, these methods may use the output of the classifier to select the best subset of features. For example, the weights assigned to different inputs (features) in logistic regression or neural networks may be used to rank them, and select the best subset among them.

2.2. Feature Selection Using a Genetic Algorithm

A GA is grounded in the concepts of biological evolution and natural selection. The principles of natural selection are believed to govern the evolution of living species. Over generations, living organisms develop characteristics that would enable them to thrive amid the adversities of their environments. A GA works much the same way, as it iteratively refines a given solution by progressively selecting better candidate solutions, while discarding inferior choices. Doing so, it mimics the mechanisms of biological evolution, namely crossover and mutations. A fitness or objective function is used to estimate the quality of each solution. A GA takes as input a set of m-dimensional vectors in R^m as given in Equation (1):

$$\{X_1^{(m)}, X_2^{(m)}, ..., X_k^{(m)}\}, \tag{1}$$

where

$$X_i^{(m)} = [x_1, x_2, ..., x_{m-1}, x_m]. \tag{2}$$

The GA generates an n-dimensional subset of vectors in R^n, as given in Equation (3):

$$\{X_1^{(n)}, X_2^{(n)}, ..., X_k^{(n)}\}, \tag{3}$$

where

$$X_i^{(n)} = [x_1, x_2, ..., x_{n-1}, x_n]. \tag{4}$$

The GA just reduces the dimensionality of each vector in the set given in Equation (1) without affecting the cardinality of the set, i.e., $n \ll m$. The dimensions selected by the GA are basically those SMART attributes, which minimize the fitness function. Thus, $X_i^{(n)}$ represents a vector selected by the GA in the optimal subset of selected feature vectors. Equation (5) shows the fitness function proposed in this study:

$$F = \frac{\overline{C}}{\overline{S}}, \tag{5}$$

Here, $\overline{C}$ denotes the average compactness of the classes, as given in Equation (6),

$$\overline{C} = \frac{1}{L}\sum_{i}^{L} C_i, \tag{6}$$

and $\overline{S}$, as given in Equation (7):

$$\overline{S} = \frac{2}{L(L-1)}\sum_{i \neq j}^{L} S_{ij}, \tag{7}$$

represents the average value of the separability among different classes, while L is the total number of classes in a given problem. In this study, the number of classes is two, i.e., healthy and failed hard disks.

The notions of compactness of a class and the separability of two classes are illustrated in Figure 2 in a three-dimensional space. The compactness of a class measures how well different instances of that class are clustered together, whereas separability between two classes estimates how easy it is to separate the clusters formed by the instances of those two classes. The proposed fitness function is intuitively similar to the notion of Fisher ratio and, in some studies, it has been used in combination with Fisher ratio for feature selection [34]. The GA iterates to find a subset of features that would minimize the fitness function, as given in Equation (5), which is a ratio of the average values of these two quantities.

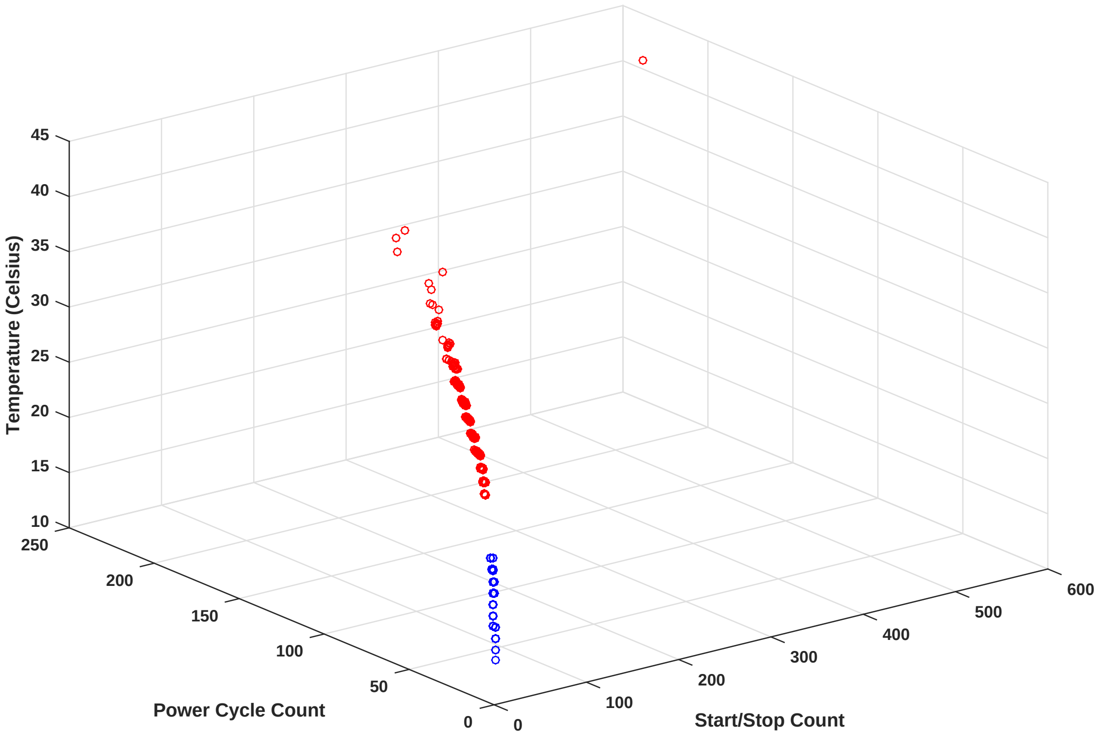

Figure 2. The distribution of three SMART attributes for healthy (blue) and failed (red) HDD.

The compactness of a given class is estimated by calculating the distance, mostly the Euclidean distance, between the feature vectors of the class and the class mean or centroid. The mean or centroid, $\mu^{(i)}$, of a class i, which has a total of N instances, can be determined using Equation (8):

$$\mu^{(i)} = \frac{1}{N}\sum_{j=1}^{N} X_j. \tag{8}$$

The mean value of the Euclidean norm given in Equation (9), which measures the distance of an instance of the class from the class centroid, can be used as an estimate for the compactness of class i,

$$C_i = \frac{1}{N}\sum_{j=1}^{N} \|X_j - \mu^{(i)}\|. \tag{9}$$

The Euclidean distance between the centroids of classes i and j, as given in Equation (10), can be used as an estimate for the separability between these two classes:

$$S_{ij} = \|\mu^{(i)} - \mu^{(j)}\|. \tag{10}$$

The proposed two-tier approach for feature selection using a GA and feature significance scores is illustrated in Figure 3. The GA requires the SMART attributes or features to be encoded into chromosomes. New chromosomes are generated from old ones using the mechanism of mutations and crossovers. The newly generated chromosomes replace their parents, provided that they are better, i.e., they perform better compared to their parents in terms of the fitness or objective function, as discussed earlier. This process of generating new chromosomes and selecting the best among them to replace the ones in the last generation is iterated multiple times until the fitness function reaches an asymptotic value with further iterations yielding no improvement in the fitness function [27].

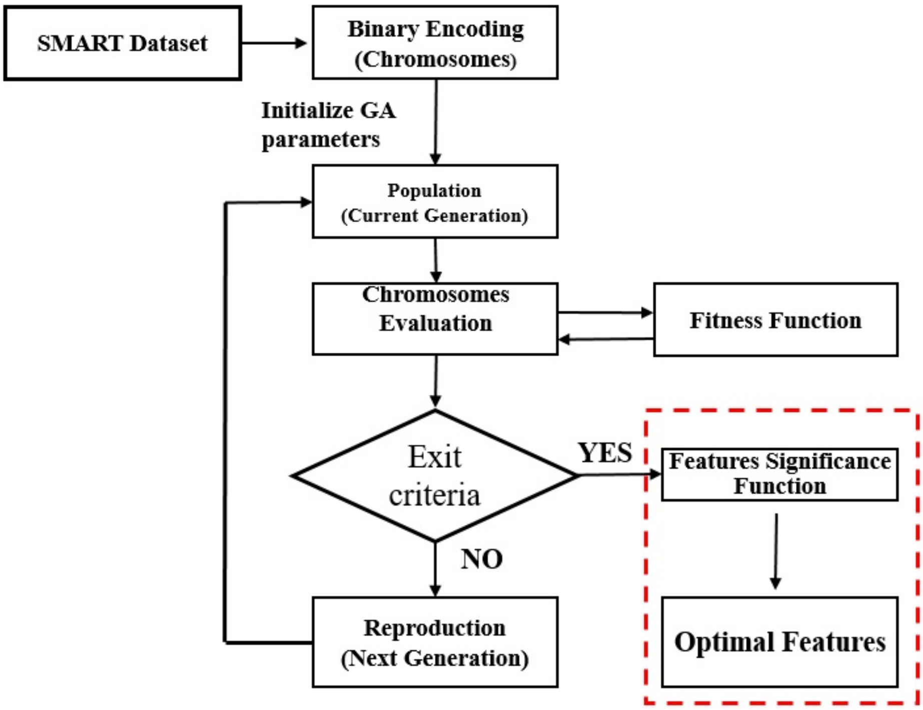
SMART Dataset
Binary Encoding (Chromosomes)
Initialize GA parameters
Population (Current Generation)
Chromosomes Evaluation
Fitness Function
Exit criteria
YES
NO
Features Significance Function
Optimal Features
Reproduction (Next Generation)

Figure 3. The flow of the proposed feature selection algorithm using a genetic algorithm and feature significance scores.

The SMART attributes are encoded into chromosomes using a binary encoding scheme. The resulting chromosomes are nothing but strings of $1's$ and $0's$, where 0 implies that a certain SMART attribute has not been selected and 1 implies that it has been selected. A binary encoding scheme is used because we want to either select or leave a particular feature. When a particular feature is selected, only then is it used in the calculation of the fitness function. However, if a given feature is not selected, then it is left out completely in the calculation of the fitness function. The binary encoding scheme is preferred here as opposed to, say, value encoding because we are not interested in determining the weights that need to be assigned to individual features; rather, we want to determine the best subset of features that will optimize the fitness function. The individual SMART attributes are identified by the indices of each 1 and 0 in each chromosome. A random string of $1's$ and $0's$, i.e., a randomly selected subset of SMART attributes serves as the initial population for the Genetic Algorithm. New populations are generated from these chromosomes by making them mutate and crossover among each other. A crossover involves the exchange of information or swapping of fragments between two parent chromosomes at randomly selected points. In contrast, mutations involve the flipping of bits on a single chromosome at randomly selected positions. To select the best set of chromosomes that could be used to replace the old ones, the value of the fitness function is calculated for each chromosome in the new generation of chromosomes. The first 100 chromosomes, which have the smallest fitness function values (100 is the size of the chromosome population in this study), are selected to produce the next generation of chromosomes. This process of creation and selection goes on for multiple generations until the proposed fitness function reaches an asymptotic value and sees no further reduction.

2.3. Feature Selection Using Significance Scores

The GA evaluates subsets of features or SMART attributes in high-dimensional spaces. Given the conclusions drawn in previous studies, such as [16], which is summarized in Section 1, it is plausible to assume that these features might behave well as a group, i.e., lead to easily distinguishable clusters in the feature space when considered together. However, individually, they might not hold considerable significance in determining a drive's failure. Hence, a simple mechanism is proposed to individually evaluate each feature selected by the GA by calculating its significance score, which provides a crude measure of the contribution of that feature towards the failure of disk drives. For a given feature, x_i, first its mean values for both healthy and failed disk drives are determined, denoted as $\overline{x_i}^{(h)}$ and $\overline{x_i}^{(f)}$, respectively. Then, the significance score Ψ is determined by calculating the frequentist probability of the event, when the distance of a sample of feature x_i from $\overline{x_i}^{(f)}$ is smaller than its distance from $\overline{x_i}^{(h)}$, as given in Equation (11):

$$\Psi = P(\|x_i - \overline{x_i}^{(f)}\| < \|x_i - \overline{x_i}^{(h)}\|) \tag{11}$$

According to Equation (11), a feature is considered significant for predicting a disk failure if most of its values for failed drives lie closer to the mean value of that feature for the failed disk drives.

2.4. Classification Using the Naive Bayes Classifier

In this study, the NB classifier is used to differentiate a healthy HDD from a failing drive using the features selected by the proposed two-tier feature selection method, which employs a GA and feature significance scores. The NB classifier classifies a given feature vector, X_n, by applying Bayes' rule to a generative classifier of the form given as follows [35]:

$$P(y = c|X_n, \theta) \propto P(X_n|y = c, \theta)P(y = c|\theta) \tag{12}$$

In Equation (12), $y \in \{H, F\}$ is a class label and θ is the unknown parameter for the conditional density of each class, while H and F represent healthy and failing hard disks, respectively. Unlike discriminative classifiers such as support vector machines (SVMs) that directly model the posterior $P(y|X_n)$, generative classifiers such as NB solve a more general problem, i.e., they learn a model of the joint probability, $P(X, y)$ of the feature vectors X and class label y and then use Bayes' rule to predict the label for an unknown feature vector [36]. The NB classifier assumes that there is conditional independence among the features, given the class labels, which reduces the number of unknown parameters that need to be estimated [35]. This is a reasonable assumption, given that, in the SMART dataset [5], most of the features are independent and do not generally exhibit any correlation. The NB classifier predicts categorical class labels, i.e., Healthy and Failing, for unknown vectors of the selected SMART attributes by using the Bayes rule and estimating the joint probability of the SMART attributes and the class labels. Given the nature of SMART data, a direct mapping of the feature vectors and the output labels by a discriminative classifier, such as an SVM, would not be very useful in predicting a failing disk drive. Moreover, because of the inherent class imbalance in the available data, i.e., the ratio of failed to healthy drives being a little more than 1:100, discriminative learners can lead to overfitting, and may have difficulty learning the minority class distribution [37–40]. This is demonstrated by the experimental results as discussed in Section 4. The model is constructed using a multivariate multinomial distribution (MVMN), which fits an appropriate probability distribution model for each attribute. A three-fold cross-validation scheme is used to improve the generalization performance of the NB classifier by training and testing it on different subsets of the data.

3. The SMART Dataset

The proposed methodology for detecting failing hard disks was tested on a public dataset of SMART attributes that was collected by a commercial datacenter [5]. The dataset used in this study was collected for more than 40, 000 HDD, comprising 26 different models and with storage capacities ranging from 1.0 terabyte to 8.0 terabytes, over a span of 273 days. As disks that were determined to have failed on a given day were removed the next day, and new disks were constantly being added to the datacenter, the number of disks was almost always changing, though it remained in excess of

40, 000 at all times. However, the proposed methodology wasn't tested on the entire dataset because of several caveats in the data. First, although values of 45 SMART attributes were to be recorded for more than 40, 000 HDD each day, not every HDD reported values for all the SMART attributes every day; a few attributes weren't reported at all. Therefore, the data were rife with blank fields, requiring the removal of records for disks with blank fields. Moreover, for some SMART attributes, the reported values were way out of bounds, e.g., for some drives, the raw value of SMART attribute "9" (9 is the ID of the SMART attribute that stores the *total power-on hours* for a HDD) suggested a drive life of more than 10 years, which was not correct. Hence, such records had to be removed from the dataset to ensure that the results and conclusions are based on reliable data. Furthermore, an average ARR of around 2–4% implies that the available SMART datasets would almost always be imbalanced, i.e., more data are available for healthy drives than for failed drives; class imbalance skews the classifiers in favor of the majority class, i.e., the healthy drives in this case, which leads to overfitting. Different approaches have been proposed to mitigate the effects of class imbalance. The simplest approaches involve either over-sampling the minority class, i.e., the failed hard disks in this case, or under-sampling the majority class, i.e., the healthy disks [37–40]. In this study, the majority class is under-sampled, i.e., data for an appropriate number of randomly selected healthy hard disks are removed to make the two classes more balanced. The SMART attribute values for all the failed drives are selected for training the NB classifier, i.e., a total of 565 drives failed during the period for which the data were considered. In order to have a reasonable balance between the two classes, SMART data for only 1500 randomly selected samples of healthy hard drives were added to the final dataset, which reduced training time and helped avoid overfitting. Hence, the final dataset that was used for training and testing the classifier contained values of 42 SMART attributes for 2065 hard disks, for a period of approximately nine months.

4. Results and Discussion

This section presents the results of this study, and provides a discussion in the context of existing research. After the necessary preprocessing, as discussed in Section 3, the SMART data are used by the GA to select the optimal features that will minimize the proposed fitness function. The subset of features selected by the GA is given in Table 1. The GA reduces the dimensionality of the original feature space from 42 to 12. It utilizes a population of 100 chromosomes over a maximum of 80 generations to optimize the fitness function discussed in Section 2.

Table 1. Features selected by the Genetic Algorithm only.

S. No.	SMART ID	Attribute Name
1	3	Spin-up Time
2	4	Start/Stop Count
3	7	Seek Error Rate
4	10	Spin Retry Count
5	12	Power Cycle Count
6	187	Reported Uncorrected Errors
7	189	High Fly Writes
8	193	Load/Unload Cycle Count
9	194	Temperature
10	197	Current Pending Sector Count
11	198	Uncorrectable Sector Count
12	199	UltraDMA CRC Error Count

The GA selects a subset of features, which might work well as a combination but might contain sub-optimal features. Those sub-optimal features might not be significant contributors in determining whether an HDD is failing or not. This is why, in the proposed two-tier feature selection process, all the features selected by the GA are subsequently evaluated by calculating their feature significance scores, as discussed in Section 2.3.

The second tier of the proposed feature selection process discards the features at serial numbers 1, 4, and 6, which are the SMART attributes recording the *spin-up time*, *spin retry count*, and *reported uncorrectable errors*, respectively. The final list of the nine features selected by the proposed two-tier feature selection process is given in Table 2. The effectiveness of the proposed two-tier feature selection process is demonstrated by the results in Table 3. When an NB classifier is trained using the features selected by the proposed two-tier approach, given in Table 2, it achieves an average classification accuracy of 99.01% with a false positive rate (FPR) of 0.24%. However, when the NB classifier is trained using different sets of SMART attributes, the results are not as promising. For example, when all 42 attributes are used to train the classifier, the average classification accuracy drops to 86.98%, while the FPR rises to 1.03%. The features selected by the GA alone are effective, but contain certain SMART attributes, which are not significant contributors in determining a failing disk drive, as evident from the average accuracy of 92.0% and an FAR of 0.92%. Table 3 also provides a comparison of NB with a discriminative classifier, such as an SVM. The SVM achieves an average accuracy of 83.30% and an FAR of 0.26%, when it is trained on the features selected by the GA alone. It achieves a marginal improvement in average accuracy when it is trained on the nine features selected by the proposed two-tier approach. However, it shows degradation in both the TPR and FPR, i.e., the TPR drops to 44%, whereas the FPR exhibits a small increase of 0.6%. The results in Table 3 indicate that the NB classifier performs better than SVM in predicting failing disk drives. Moreover, the results obtained using two different types of classifiers indicate that the SMART attributes selected by the proposed two-tier feature selection method yield better diagnostic performance in detecting failing and healthy HDD.

Table 2. Features selected by the proposed two-tier feature selection process.

S. No.	SMART ID	Attribute Name
1	4	Start/Stop Count
2	7	Seek Error Rate
3	12	Power Cycle Count
4	189	High Fly Writes
5	193	Load/Unload Cycle Count
6	194	Temperature
7	197	Current Pending Sector Count
8	198	Uncorrectable Sector Count
9	199	UltraDMA CRC Error Count

Table 3. A comparison of the performance achieved by the Naive Bayes and Support Vector Machine classifiers using different feature selection methods.

Method	Feature Vector Dimensionality	No. of Folds for Cross Validation	No. of Test Iterations	False Positive Rate (%)	True Positive Rate (%)	Average Accuracy (%)
Naive Bayes with No Feature Selection	42	3	10	1.03	55.2	86.98
Naive Bayes with GA only	12	3	10	0.92	72.0	92.0
Naive Bayes with Proposed Two-Tier Method	9	3	10	0.24	98.4	99.01
SVM with No Feature Selection	42	3	10	74.95	75.0	20.56
SVM with GA only	12	3	10	0.26	40.0	83.3
SVM with Proposed Two-Tier Method	9	3	10	0.6	44.15	84.3

As discussed earlier, the SVM, being a discriminative classifier, models the direct relationship between the feature vectors and the class labels. This approach may not be effective in the case of SMART data, where a direct relation between the two is not always very conclusive. The performances of the SVM and NB classifiers are also illustrated in Figure 4, which compares the receiver operating characteristic (ROC) curves for these classifiers.

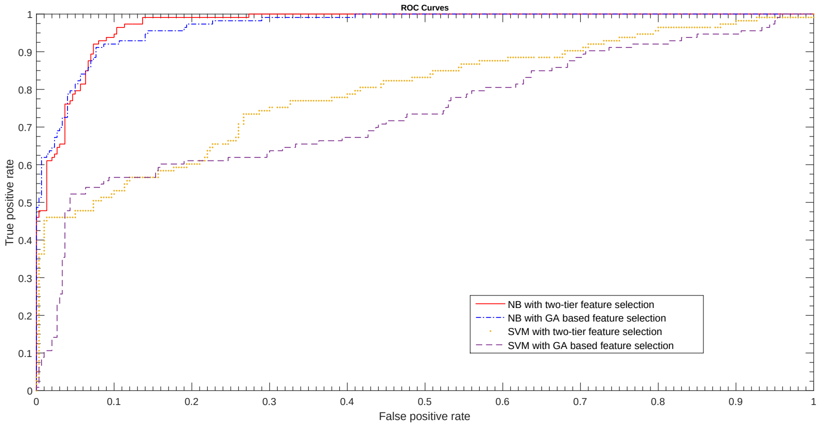

Figure 4. The receiver operating characteristic (ROC) curves for SVM and Naive Bayes Classifiers using different feature selection schemes.

The ROC curves in Figure 4 clearly indicate that the NB classifier trained on the nine features selected by the proposed two-tier feature selection process yields the best results in terms of both the TPR and FPR. When compared to existing methods for detecting failing hard disks, the proposed method offers distinct advantages. Wang et al. [18] presented a two-step parametric method, which utilizes 47 critical features, identified in [17] for failure prediction in HDD, as opposed to the nine SMART attributes determined in this study. They tested their method on a dataset that was collected for 369 hard disks of a single model, and contained data only for the last 600 h, where each sample of data are 2 h apart from the next one. Thus, for each drive, a maximum of 300 values are available for each of the 47 features. Among the 369 drives, 178 drives are healthy, while 191 are failed drives. Given the variation in failure rates across different models and across different storage capacities for certain models, as observed in commercial datacenters [41], this dataset cannot be considered a representative dataset. In contrast, as discussed in Section 2, the proposed method is tested on a more extensive and more representative dataset. The method proposed in [18] could achieve a failure detection rate (FDR) or TPR of 68.42% at a FAR of 0%, and an FDR of around 95% at a FAR of around 4.2%. However, the FAR is highly sensitive to the failure threshold and no mechanism has been provided to set an appropriate failure threshold. This is an important concern because the ineffectiveness of the simple thresholding technique put in place by drive manufacturers to detect a failing hard disk was a major reason why considerable interest was generated among researchers to devise better methods for detecting failing disk drives. The proposed method uses a two-tier feature selection process, and determines the nine SMART attributes given in Table 2, to be the most effective precursors to a failing HDD. This reduces the training time of the classifier. Using the values of the nine selected SMART attributes for 2065 hard disks, the proposed method yields an FDR of 98.40% at a FAR of 0.24%, without using any arbitrary parameters. Among the 2065 hard drives, 1500 are healthy and 565 are failed drives. These are divided into three folds, with 688, 688, and 689 hard drives, respectively. Each fold contains 500 healthy drives, whereas the remainder are failed drives. The proposed method correctly detects 185 of the 188 failed drives with only 1.2 false alarms on average. Moreover, the proposed algorithm can also be used for the online diagnosis of HDD. A trained instance of a NB or SVM classifier can be provided with the values of the nine SMART attributes of an HDD, as listed in Table 2. The output of the classifier can then be interpreted as either a healthy HDD or one with an impending failure.

5. Conclusions

In this paper, a novel two-tier approach was presented to select the most effective precursors to a failing HDD. These precursors were selected from an initial list of 42 SMART attributes, which were recorded in a commercial datacenter over a period of nine months for 21 different models with storage capacities ranging from 1.0 TB to 8.0 TB. The proposed two-tier approach evaluated the SMART attributes, both in combinations and individually. First, a GA was used to explore different feature subspaces in order to determine the best combination of features or SMART attributes. The quality of a feature subset was measured by determining how well clustered the samples of those features are for each of the two classes, i.e., healthy and failed HDD, and how well separated those clusters are from one another. This was done by calculating the ratio of intra-class compactness to inter-class separation for each subset of features, and finding the subset that minimized this ratio. The compactness of a class and the separation between two classes were measured using Euclidean distances. In the second tier, a new measure, i.e., significance score, was proposed to individually evaluate the features selected by the GA. The significance score measured the statistical contribution of a given feature towards disk failures. Features with statistical scores lower than a certain value were discarded. The final list of features selected by the proposed two-tier feature selection process comprised nine SMART attributes as opposed to the original 42, resulting in a shorter training time for the classifiers. These nine attributes were then used to train a generative classifier, the NB. The NB gave an FDR of 98.40% compared to 40.0% by the SVM, which is a discriminative classifier. To avoid overfitting by the classifier on the majority class data, the inherent class-imbalance problem in the failure data for HDD was addressed by under-sampling the majority class of healthy HDD. The proposed method correctly detected 185 of the 188 failed drives with only 1.2 false alarms on average.

Author Contributions: W.A. was involved in the conception of the idea and its implementation. S.A.K. was involved in the design of experiments. J.-M.K. and C.H.K. were involved in the analysis and interpretation of results. W.A. and S.A.K. prepared the manuscript. J.-M.K. and C.H.K. were involved in the revision of the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by 2020 Research Fund of University of Ulsan.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AE	Acoustic Emission
AFR	Annual failure rate
ARR	Annual (disk) replacement rate
BIOS	Basic input output system
CRC	Cyclic redundancy check
DFP	Disk failure prediction
DMA	Direct memory access
FAR	False alarm rate
FC	Fiber channel
FDR	Failure detection rate
FPR	False positive rate
GA	Genetic algorithm
HDD	Hard disk drive
IT	Information technology
MTTF	Mean time to failure
MVMN	Multi-variate multi-nomial
NB	Naive Bayes
SATA	Serial advanced technology attachment
SCSI	Small computer system interface
SMART	Self monitoring, analysis and reporting technology
SVM	Support vector machine
ROC	Receiver operating characteristic
TPR	True positive rate

References

1. Coughlin, T. Near and Far—Digital Storage Supporting Today's Mobile Devices [The Art of Storage]. *IEEE Consum. Electron. Mag.* **2014**, *3*, 64–67. [CrossRef]
2. Gantz, J.; Reinsel, D. The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east. *IDC IView IDC Anal. Future* **2012**, *2007*, 1–16.
3. Schroeder, B.; Gibson, G.A. Understanding disk failure rates: What does an MTTF of 1,000,000 hours mean to you? *ACM Trans. Storage (TOS)* **2007**, *3*, 8-es. [CrossRef]
4. Schroeder, B.; Gibson, G.A. Understanding failures in petascale computers. *J. Phys. Conf. Ser.* **2007**, *78*, 012022. [CrossRef]
5. Hard Drive Data and Stats Volume Q1–Q3 2015. Available online: https://www.backblaze.com/b2/hard-disk-test-data.html (accessed on 30 October 2016).
6. Schroeder, B.; Gibson, G.A. Disk failures in the real world: What does an MTTF of 1, 000, 000 hours mean to you? In *FAST*; USENIX: San Hose, CA, USA, 2007; Volume 7, pp. 1–16.
7. Sankar, S.; Shaw, M.; Vaid, K.; Gurumurthi, S. Datacenter scale evaluation of the impact of temperature on hard disk drive failures. *ACM Trans. Storage (TOS)* **2013**, *9*, 1–24. [CrossRef]
8. Wang, G.; Zhang, L.; Xu, W. What can we learn from four years of data center hardware failures? In Proceedings of the 2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Denver, CO, USA, 26–29 June 2017; pp. 25–36.
9. Kamarthi, S.; Zeid, A.; Bagul, Y. Assessement of current health of hard disk drives. In Proceedings of the 2009 IEEE International Conference on Automation Science and Engineering, Vancouver, BC, Canada, 22–26 August 2009; pp. 246–249.
10. Jiang, W.; Hu, C.; Zhou, Y.; Kanevsky, A. Are disks the dominant contributor for storage failures? A comprehensive study of storage subsystem failure characteristics. *ACM Trans. Storage (TOS)* **2008**, *4*, 1–25. [CrossRef]
11. Ottem, E.; Plummer, J. *Playing It SMART: The Emergence of Reliability Prediction Technology*; Technical Report, Technical Report, Seagate Technology Paper; Seagate Technology: Scotts Valley, CA, USA, 1995.
12. Hamerly, G.; Elkan, C. Bayesian approaches to failure prediction for disk drives. In Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, 28 June–1 July 2001; Volume 1, pp. 202–209.
13. Hughes, G.F.; Murray, J.F.; Kreutz-Delgado, K.; Elkan, C. Improved disk-drive failure warnings. *IEEE Trans. Reliab.* **2002**, *51*, 350–357. [CrossRef]
14. Murray, J.F.; Hughes, G.F.; Kreutz-Delgado, K. Machine learning methods for predicting failures in hard drives: A multiple-instance application. *J. Mach. Learn. Res.* **2005**, *6*, 783–816.
15. Henry, R.K. Monitoring PC Hardware Sounds in Linux Systems Using the Daubechies D4 Wavelet. Master's Thesis, East Tennessee State University, Tennessee, TN, USA, 2005.
16. Pinheiro, E.; Weber, W.D.; Barroso, L.A. *Failure Trends in a Large Disk Drive Population*; USENIX: San Hose, CA, USA, 2007.
17. Wang, Y.; Miao, Q.; Pecht, M. Health monitoring of hard disk drive based on Mahalanobis distance. In Proceedings of the 2011 Prognostics and System Health Managment Conference, Shenzhen, China, 24–25 May 2011; pp. 1–8.
18. Wang, Y.; Ma, E.W.; Chow, T.W.; Tsui, K.L. A two-step parametric method for failure prediction in hard disk drives. *IEEE Trans. Ind. Inform.* **2013**, *10*, 419–430. [CrossRef]
19. Qian, J.; Skelton, S.; Moore, J.; Jiang, H. P3: Priority based proactive prediction for soon-to-fail disks. In Proceedings of the 2015 IEEE International Conference on Networking, Architecture and Storage (NAS), Boston, MA, USA, 6–7 August 2015; pp. 81–86.
20. Botezatu, M.M.; Giurgiu, I.; Bogojeska, J.; Wiesmann, D. 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, 13–17 August 2016; pp. 39–48.
21. Zhang, S.; Bahrampour, S.; Ramakrishnan, N.; Shah, M. Deep Symbolic Representation Learning for Heterogeneous Time-series Classification. *arXiv* **2016**, arXiv:1612.01254.

22. Black, R.; Donnelly, A.; Harper, D.; Ogus, A.; Rowstron, A. Feeding the pelican: Using archival hard drives for cold storage racks. In Proceedings of the 8th {USENIX} Workshop on Hot Topics in Storage and File Systems (HotStorage 16), Denver, CO, USA, 20–21 June 2016.
23. Zhang, T.; Wang, E.; Zhang, D. Predicting failures in hard drivers based on isolation forest algorithm using sliding window. *J. Phys. Conf. Ser.* **2019**, *1187*, 042084. [CrossRef]
24. Huang, S.; Liang, S.; Fu, S.; Shi, W.; Tiwari, D.; Chen, H.B. Characterizing disk health degradation and proactively protecting against disk failures for reliable storage systems. In Proceedings of the 2019 IEEE International Conference on Autonomic Computing (ICAC), Umea, Sweden, 16–20 June 2019; pp. 157–166.
25. Cantu-Paz, E. Feature subset selection, class separability, and genetic algorithms. In Proceedings of the Genetic and Evolutionary Computation Conference, Seattle, WA, USA, 26–30 June 2004; pp. 959–970.
26. Saberi, M. Feature selection method using genetic algorithm for the classification of small and high dimension data. *Proc. Int. Symp. Info. Com. Tech.* **2004**, 13–16. [CrossRef]
27. Min, S.H.; Lee, J.; Han, I. Hybrid genetic algorithms and support vector machines for bankruptcy prediction. *Expert Syst. Appl.* **2006**, *31*, 652–660. [CrossRef]
28. Chandrashekar, G.; Sahin, F. A survey on feature selection methods. *Comput. Electr. Eng.* **2014**, *40*, 16–28. [CrossRef]
29. Khalid, S.; Khalil, T.; Nasreen, S. A survey of feature selection and feature extraction techniques in machine learning. In Proceedings of the 2014 Science and Information Conference, Marrakesh, Morocco, 28–30 May 2014; pp. 372–378.
30. Rida, I.; Al-Maadeed, N.; Al-Maadeed, S.; Bakshi, S. A comprehensive overview of feature representation for biometric recognition. In *Multimedia Tools and Applications*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 1–24.
31. Rida, I.; Al Maadeed, S.; Bouridane, A. Unsupervised feature selection method for improved human gait recognition. In Proceedings of the 2015 23rd European Signal Processing Conference (EUSIPCO), Nice, France, 31 August–4 September 2015; pp. 1128–1132.
32. Chen, Y.; Li, Y.; Cheng, X.Q.; Guo, L. Survey and taxonomy of feature selection algorithms in intrusion detection system. In Proceedings of the International Conference on Information Security and Cryptology, Busan, Korea, 30 November–1 December 2006; pp. 153–167.
33. Rida, I.; Boubchir, L.; Al-Maadeed, N.; Al-Maadeed, S.; Bouridane, A. Robust model-free gait recognition by statistical dependency feature selection and globality-locality preserving projections. In Proceedings of the 2016 39th International Conference on Telecommunications and Signal Processing (TSP), Vienna, Austria, 27–29 June 2016; pp. 652–655.
34. Okimoto, L.C.; Lorena, A.C. Data complexity measures in feature selection. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–8.
35. Murphy, K.P. *Machine Learning: A Probabilistic Perspective*; MIT Press: Cambridge, MA, USA, 2012.
36. Ng, A.Y.; Jordan, M.I. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 9–14 December 2002; pp. 841–848.
37. Japkowicz, N. The class imbalance problem: Significance and strategies. In Proceedings of the International Conference on Artificial Intelligence, Melbourne, Australia, 28 August–1 September 2000; Volume 1, pp. 111–117.
38. Chawla, N.V.; Japkowicz, N.; Kotcz, A. Special issue on learning from imbalanced data sets. *ACM SIGKDD Explor. Newsl.* **2004**, *6*, 1–6. [CrossRef]
39. Batista, G.E.; Prati, R.C.; Monard, M.C. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explor. Newsl.* **2004**, *6*, 20–29. [CrossRef]
40. Liu, X.Y.; Wu, J.; Zhou, Z.H. Exploratory undersampling for class-imbalance learning. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* **2008**, *39*, 539–550.
41. Klein, A. Backblaze Hard Drive Stats for 2016. Available online: https://www.backblaze.com/blog/hard-drive-benchmark-stats-2016/ (accessed on 13 April 2017).