Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.067759

ARTICLE

HDFPM: A Heterogeneous Disk Failure Prediction Method Based on Time Series Features

Zhongrui Jing[1], **Hongzhang Yang**[1,*] **and Jiangpu Guo**[2]

[1]School of Computer Science and Engineering, Tianjin University of Technology, Tianjin, 300384, China
[2]R&D Department, Roycom Information Technology Corporation, Tianjin, 301721, China

*Corresponding Author: Hongzhang Yang. Email: yanghongzhang@email.tjut.edu.cn

Received: 12 May 2025; Accepted: 22 October 2025; Published: 09 December 2025

ABSTRACT: Hard disk drives (HDDs) serve as the primary storage devices in modern data centers. Once a failure occurs, it often leads to severe data loss, significantly degrading the reliability of storage systems. Numerous studies have proposed machine learning-based HDD failure prediction models. However, the Self-Monitoring, Analysis, and Reporting Technology (SMART) attributes differ across HDD manufacturers. We define hard drives of the same brand and model as homogeneous HDD groups, and those from different brands or models as heterogeneous HDD groups. In practical engineering scenarios, a data center is often composed of a heterogeneous population of HDDs, spanning multiple vendors and models. Existing research predominantly focuses on homogeneous datasets, ignoring the model's generalization capability across heterogeneous HDDs. As a result, HDD models with limited samples often suffer from poor training effectiveness and prediction performance. To address this issue, we investigate generalizable SMART predictors across heterogeneous HDD groups. By extracting time-series features within a fixed sliding time window, we propose a Heterogeneous Disk Failure Prediction Method based on Time Series Features (HDFPM) framework. This method is adaptable to HDD models with limited sample sizes, thereby enhancing its applicability and robustness across diverse drive populations. Experimental results show that the proposed model achieves an F1-score of 0.9518 when applied to two different Seagate HDD models, while maintaining the False Positive Rate (FPR) below 1%. After incorporating the Complexity-Ratio Dynamic Time Warping (CDTW) based feature enhancement method, the best prediction model achieves a True Positive Rate (TPR) of up to 0.93 between the two models. For next-day failure prediction across various Seagate models, the model achieves an F1-score of up to 0.8792. Moreover, the experimental results also show that within the same brand, the higher the proportion of shared SMART attributes across different models, the better the prediction performance. In addition, HDFPM demonstrates the best stability and most significant performance in heterogeneous environments.

KEYWORDS: Heterogeneous hard disk drives; failure prediction; time series feature; constrained dynamic time warping; sensitivity analysis

1 Introduction

In large-scale storage scenarios, such as high-performance computing (HPC) and internet-based services, frequent read/write operations make HDD failures increasingly common. According to the investigation in [1], approximately 78% of hardware replacements are caused by HDD failures. Such failures can lead to data loss, increased operational costs, and significant economic impact, ultimately reducing the overall reliability of data centers. SMART is a technology that can detect various operational parameters of HDDs, such as read/write counts, head load/unload cycles, seek error rates, and more. In past research,

many scholars have proposed various methods using machine learning and deep learning techniques to improve failure prediction accuracy, including the use of Long Short-Term Memory (LSTM) [2–4] and Convolutional Neural Networks (CNNs) [5] as well as the latest Siamese Neural Network combined with attention mechanism method (SiaDFP) [6], to build hard drive failure prediction models. Despite the extensive research in this field and the achieved promising results, these studies have primarily focused on specific brands and models of HDDs, typically those with the highest number of units and failures. However, in real industrial settings, HDDs in the same data center often come from multiple brands and models. HDD failure prediction in heterogeneous environments remains an open issue, resulting in difficulty in predicting and training models for HDDs types with limited data.

To address this issue, we conduct a large-scale analysis of hard drive failures in heterogeneous environments and introduce the HDFPM model for the first time. We process the hard drive monitoring data (SMART attributes) using CDTW and then employ the HDFPM model for prediction. This model demonstrates effective prediction of hard drive failure in the context of heterogeneous environments. The main contributions of this paper are as follows:

We propose the HDFPM model, which is tailored for hard drive failure prediction in heterogeneous HDD environments. This study utilizes the SMART data of all HDDs published by Backblaze in Q1 2023, combined with HDD data collected by Roycom Information Technology Corporation during the same period. Four datasets with different levels of heterogeneity are constructed for evaluation. In addition, a sensitivity analysis of the sampling strategy is also conducted.

Our study incorporates time-series features, which leverage the temporal differences in SMART attributes between healthy and failed hard drives. For each selected SMART attribute, CDTW is applied to enhance the extracted features, enabling effective measurement of variations between adjacent observations in the SMART time series.

We investigate the optimal time prediction window based on the behavior of time-series features.

We investigate the impact of shared attributes between heterogeneous models within the same brand on the prediction performance.

We also conduct statistical robustness and significance analysis of the HDFPM model.

2 Related Work

In past research, many scholars have proposed machine learning and deep learning-based approaches to improve the accuracy of hard drive failure prediction. These studies primarily focus on four aspects: time-series feature selection, predictive time window forecasting, data balancing, and prediction models.

2.1 Regarding Time-Series Features

Shen et al. [7] used Euclidean transformation to calculate the distance between healthy disk samples and the last failure sample, designing a failure detection model. Huang et al. [8] studied read/write head failures and bad sector failures, using Euclidean distance to compute the differences between each SMART record and the failure record prior to the failure event. Chakraborttii et al. [9] proposed an anomaly detection method that combines a one-class isolation forest with an autoencoder to extract time-series features for inferring disk failures. Although this method was evaluated on several minority disk models, it has not been assessed across a wider range of disk models. Shirke et al. [10] conducted a systematic review of research in the field of computational storage. Hard disk failure prediction still holds great potential for future development. Han et al. [11] proposed a stream mining framework with concept-drift adaptation for extracting time-series features of hard disks. The framework supports online prediction and training on the

generated disk data streams. Although it achieved good performance, it still suffered from a high false positive rate. However, the above studies are all focused on hard drives of specific brands and models, resulting in limited generalization capability.

2.2 Regarding Predictive Time Windows

Botezatu et al. [12] introduced a changepoint-based feature selection method to perform high-precision analysis on SMART indicators and represent them in a compact form. Combined with the RGF (Regularized Greedy Forest) classifier, their approach can predict hard drive failures 10–15 days in advance. Li et al. [13] combined SMART data from hard drives and employed XGBoost, LSTM, and ensemble learning to predict hard drive failures. This method is capable of predicting failures up to 42 days in advance. Zhang et al. [14] proposed a lookahead-window-constrained model based on a dynamic sample relabeling method that incorporates lookahead time constraints and failure symptom duration to enhance the model's predictability. Furthermore, dynamic weighted optimization was introduced during backpropagation to further improve prediction accuracy. Züfle et al. [15] employed a Random Forest-based approach for hard disk failure prediction, with a prediction window of seven days.

2.3 Regarding Data Imbalance

Kaur et al. [16] proposed a data balancing strategy based on the Binary Sample Alignment (BSA) algorithm for positive and negative sample matching. Burrello et al. [17] used the SMOTE sampling technique, a synthetic minority oversampling method, to handle the problem of class imbalance. Although this approach achieved better prediction performance, it is less effective in heterogeneous environments. Wang et al. [18] introduced class weight parameters during the training process to increase the weight of failure samples, enabling the classifier to focus more on failure samples during learning. Sun et al. [19] proposed a Temporal CNN model to mine time-series features and optimized the loss function to address the issues caused by dataset imbalance. Zhang et al. [20] addressed the data imbalance issue by proposing a Hierarchical Perturbation Neural Network, which alleviates overfitting caused by data imbalance by injecting perturbations at arbitrary layers within the network.

2.4 Regarding Prediction Models

Ensemble learning is widely used for time-series data classification and prediction. Galicia et al. [21] proposed an ensemble model based on decision trees, gradient boosting trees, and random forests for predicting large-scale time-series data. They also introduced dynamic and static prediction models based on a strategy for selecting weight coefficients. Sun et al. [22] proposed a financial time-series prediction model based on AdaBoost and Long Short-Term Memory (LSTM) networks. The model first trains the time-series data using LSTM, and subsequently integrates the results via AdaBoost, achieving good performance on nonlinear and irregular data. Yang et al. [23] utilized SMART data extracted from ZTE and employed LSTM for training and modeling, achieving good results. Gao et al. [24] selected disks with minimal differences from the target disk for feature extraction and training, applying transfer learning. This model is designed for scenarios involving a small number of disk types. Wang et al. [25] described the hard drive degradation process using a first-order Markov mixed-jump degradation model and modeled the real-time state of the hard drive by combining it with an improved Rao-Blackwellized particle filtering algorithm. Zhou et al. [26] proposed an active semi-supervised learning model for hard drive failure prediction, performing active learning on clearly labeled samples and semi-supervised learning on unclearly labeled samples. Lima et al. [27] proposed a testing framework model based on deep recurrent neural networks to predict hard drive health status. Zhang et al. [28] proposed a Multi-View Multi-Task Random Forest (MVTRF) approach, based

on data from the Samsung data center, for implementing both long-term and short-term data monitoring to predict hard drive failures.

2.5 Section Summary

The methods in the aforementioned studies learned only the SMART data distribution of specific HDD models and achieved good prediction performance within homogeneous HDD groups. However, these methods are not applicable to heterogeneous hard drive group scenarios, which motivates the research presented in this paper.

3 Datasets and Method

In this section, Section 3.1 presents the four datasets with varying degrees of heterogeneity that we collected and constructed, along with a sensitivity analysis of SMOTE sampling and oversampling by duplicating failure samples, while Section 3.2 provides an in-depth discussion on the complexity of hard drive failure prediction methods in heterogeneous environments.

3.1 Datasets and Sensitivity Analysis of Sampling Strategies

Section 3.1.1 presents the four datasets with varying degrees of heterogeneity, while Section 3.1.2 focuses on the sensitivity analysis of sampling strategies.

3.1.1 Heterogeneous Dataset

The four datasets with varying degrees of heterogeneity that we collected and constructed are presented as follows: Dataset-1 contains data from two Seagate HDD models, as shown in Fig. 1a; Dataset-2 includes 28 Seagate models, as shown in Fig. 1b; Dataset-3 consists of 41 models from two major brands, Seagate and Hitachi, as shown in Fig. 2a; and Dataset-4 covers 65 HDD models from more than 10 different vendors, as shown in Fig. 2b. These datasets provide a comprehensive basis for validating the proposed model in diverse and heterogeneous environments.

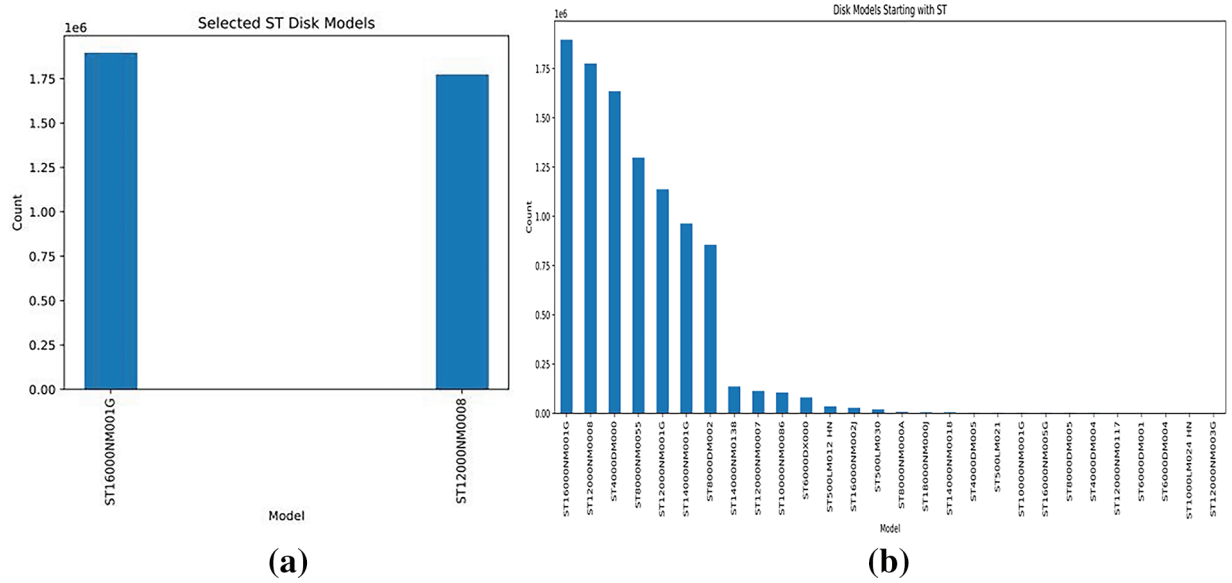


Figure 1: The cases in the two heterogeneous datasets are illustrated as follows: (**a**) Number of HDDs in Dataset-1; (**b**) Number of HDDs in Dataset-2

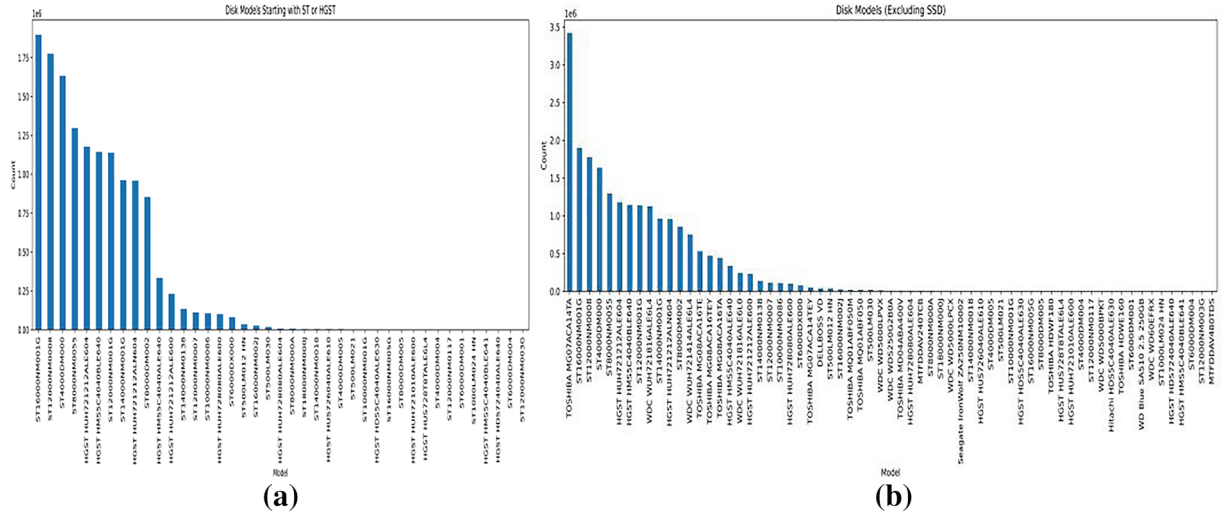


Figure 2: Apart from Fig. 1, additional cases in the heterogeneous datasets are illustrated as follows: (**a**) Number of HDDs in Dataset-3; (**b**) Number of HDDs in Dataset-4

3.1.2 Sensitivity Analysis of Sampling Strategies

To analyze the sensitivity of the oversampling strategy (i.e., duplicating failure samples) and the SMOTE sampling strategy, we conducted TPR comparison experiments on Dataset-2. The dataset was split into 70% for training and 30% for testing; the training set was processed using either simple oversampling (duplicating failure samples) or SMOTE oversampling method, while the testing set retained the original data distribution to approximate real-world scenarios as closely as possible. The proportion of failure samples was gradually increased from the original ratio up to 100% of the normal samples, and performance variations were monitored throughout the process.

The experimental results are shown in Fig. 3. Compared to oversampling, SMOTE generally achieves higher TPR values across all models on Dataset-2, and the peak performance appears at higher sampling ratios. However, as the sampling ratio increases, the performance of SMOTE fluctuates more significantly, with larger curve variations, indicating stronger sensitivity to the sampling ratio. The reason is that SMOTE generates synthetic samples of new hard drive models, and some of these synthetic samples may interfere with the patterns of the original hard drive data. In contrast, oversampling duplicates the existing failure samples, ensuring the authenticity and effectiveness of the data. Therefore, we selected the oversampling method, which shows lower sensitivity to sampling ratios and better stability.

3.2 Method

In this section, our study delves into the complexity of hard drive failure prediction methods in heterogeneous environments. Section 3.2.1 presents the overall architecture diagram and its stages; Section 3.2.2 investigates general feature selection under heterogeneous environments; Section 3.2.3 explores feature enhancement using Complexity-Ratio Dynamic Time Warping; Section 3.2.4 presents the HDFPM architecture diagram; and Section 3.2.5 investigates the impact of shared attributes across heterogeneous hard drive models on modeling.

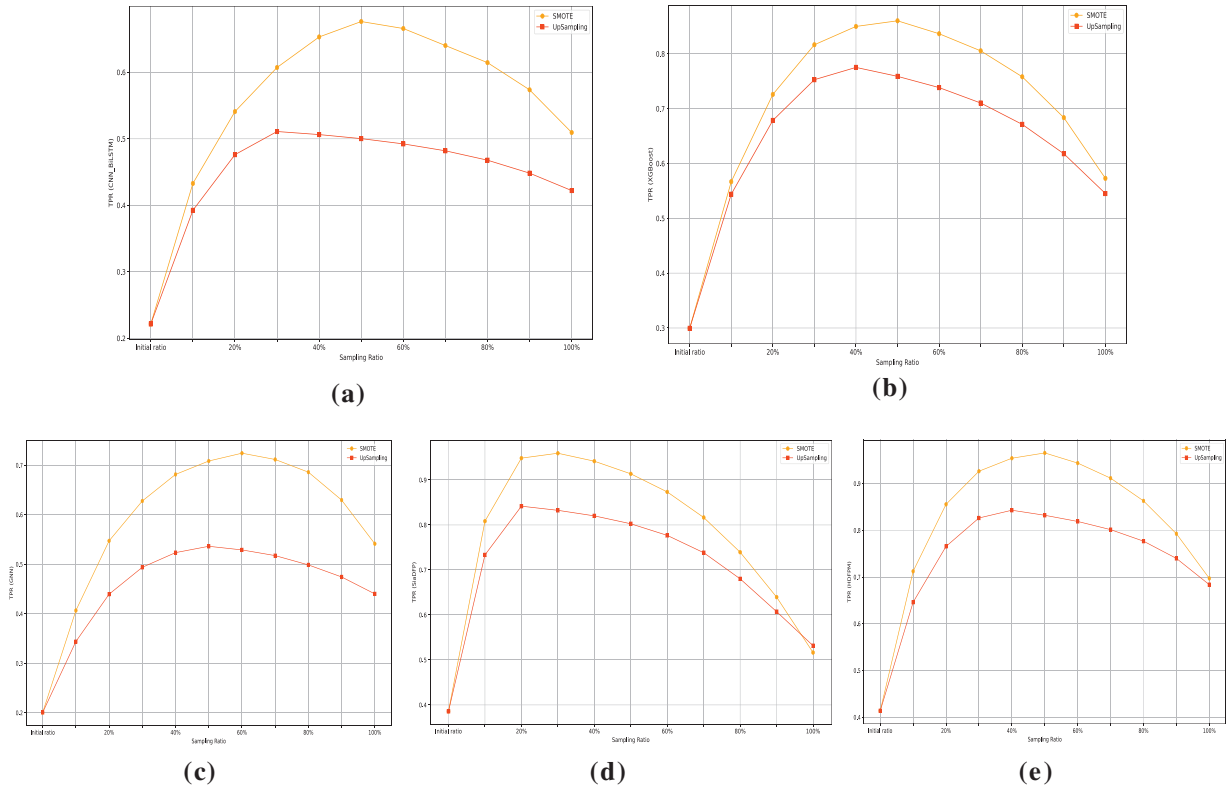


Figure 3: SMOTE and oversampling sensitivity testing results of the CNN_BiLSTM, XGBoost, GNN, SiaDFP, and HDFPM models on Dataset-2. (**a**) shows the sampling strategy sensitivity testing results of the CNN_BiLSTM model; (**b**) depicts the sampling strategy sensitivity testing results of the XGBoost model; (**c**) displays the sampling strategy sensitivity testing results of the GNN model; (**d**) shows the sampling strategy sensitivity testing results of the SiaDFP model; and (**e**) depicts the sampling strategy sensitivity testing results of the HDFPM model

3.2.1 Overall Structure

The overall architecture of this study is shown in Fig. 4, which includes four stages: (1) SMART data collection; (2) Data preprocessing, including data grouping, selection of general features in heterogeneous environments, and CDTW feature enhancement; (3) Modeling using the training dataset on a GPU server; (4) Predicting the test results based on the trained model with the test dataset. The research across these stages primarily focuses on the following aspects: the selection of general features under heterogeneous environments, feature enhancement, and the impact of brand and model differentiation on the modeling process. Additionally, this paper investigates the impact of the proportion of shared attributes on prediction performance, the sensitivity to sampling strategies, as well as the robustness and significance analysis of the model. The overall workflow is illustrated in Algorithm 1.

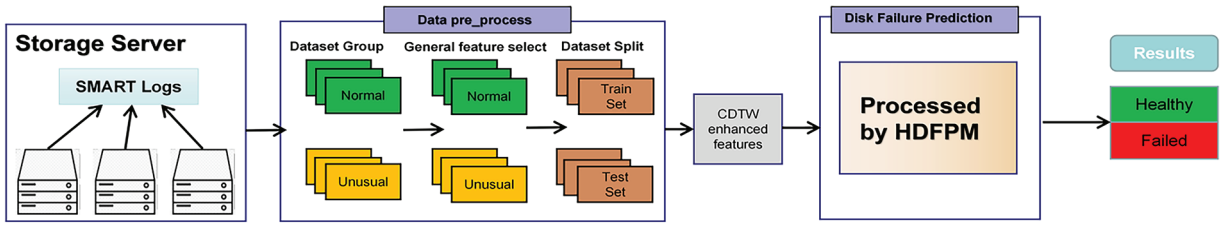


Figure 4: Overall structure

Algorithm 1: Overall workflow

1: Step 1: CollectSMARTData(DataCenter) → BuildFourHeterogeneityLevels(Datasets)
2: Step 2: SamplingSensitivityAnalysis(Datasets)
→ Analyze the influence of various sampling ratios on model performance
→ Apply oversampling methods appropriate to heterogeneous environments
3: Step 3: F_selected ← RFE(RetainImportantFeatures(VIFAnalysis(HeterogeneousFeatures)))
→ Identify highly collinear features through VIF analysis under heterogeneous environments
→ Preserve domain-significant features according to literature evidence
→ Eliminate redundant or non-essential attributes
→ Use Recursive Feature Elimination (RFE) to obtain the final optimal feature subset F_selected
4: Step 4: CompareModelPerformance(Models, Datasets_with_SelectedFeatures)
5: Step5: CompareFeatureEnhancement({None, DTW, CDTW}, Datasets_with_SelectedFeatures)
→ Apply three enhancement methods
→ Train and evaluate models under each method
→ Compare performance
6: Step 6: OptimalWindow ← FindBestPredictionWindow(Models, Datasets)
7: Step 7: AnalyzeSharedAttributeImpact(Models, Datasets)
8: Step 8: CrossValidate(Models, Datasets, folds = 10) → AnalyzeRobustnessAndSignificance()
→ Evaluate robustness via 10-fold validation
→ Perform significance testing across models

3.2.2 General Feature Selection under Heterogeneous Environments

The raw values in the SMART attributes are removed due to their large value ranges and limited contribution to modeling and prediction. In addition, high correlation between two or more attribute features may lead to biased or misleading results in the model, a phenomenon known as multicollinearity. The degree of multicollinearity can be measured using the Variance Inflation Factor (VIF) calculated for each feature in the regression set. A high VIF value indicates that the corresponding feature is highly collinear with other features in the model and should be removed. Typically, a VIF value less than 5 is considered acceptable, while a VIF value greater than or equal to 5 is considered problematic. During the feature selection process, for highly collinear features identified by VIF analysis, we prioritize the retention of features based on domain knowledge of hard drives and evidence from related literature, favoring those that are widely recognized or theoretically more critical. If multiple features are considered important, they are retained and passed into the subsequent Recursive Feature Elimination (RFE) stage for further analysis. After the initial filtering of irrelevant features, we employ the RFE method to further identify the most predictive features.

As a wrapper-based feature selection algorithm, RFE is widely adopted for its ease of implementation and its effectiveness in identifying features that are most relevant to the target variable. By complementing the limitations of VIF, RFE helps ensure the representativeness and effectiveness of the selected key features to the greatest extent. The detailed procedure is described in Algorithm 2. Table 1 presents the results of feature selection based on Dataset-1, Dataset-2, Dataset-3, and Dataset-4.

Algorithm 2: Feature selection process

Step 1: VIF-Based Multicollinearity Filtering
Construct the dataset X using all SMART attributes.
For each attribute X_i in X do:
    Compute Variance Inflation Factor (VIF_i)
End for
Remove attributes with $VIF_i \geq \theta$ (e.g., $\theta = 5$) to reduce multicollinearity
For highly collinear feature identified by VIF analysis:
    Retain features based on domain knowledge and evidence from relevant literature,
    favoring those that are widely recognized or theoretically more critical.
    If multiple features are considered important, retain all of them for the next step.
Let X_filtered be the remaining feature subset after VIF filtering.
Step 2: Recursive Feature Elimination (RFE)
Input the filtered feature subset X_filtered from Step 1.
Apply RFE to evaluate the importance of each feature in X_filtered
Select the top-N most important features to form the final feature subset X_final
Use X_final for subsequent model training and prediction

Table 1: Feature selection using RFE based on all Datasets

Number	SMART	Dataset-1	Dataset-2	Dataset-3	Dataset-4
1	Read error rate	√	√	√	√
3	Spin-up time	√			√
4	Start/Stop count	√	√	√	
5	Reallocated sectors count	√	√		
7	Seek_Error_Rate	√			
8	Seek_Time_Performance				√
9	Power-on Hours	√	√		√
10	Spin retry count	√			
188	Command timeout	√		√	
193	Load/Unload cycle count		√	√	√
194	Temperature celsius	√		√	√
200	Multi zone error rate		√	√	
240	Head flying hours		√		
241	Total LBAs written			√	

3.2.3 Feature Enhancement Using Complexity-Ratio Dynamic Time Warping

Complexity-Ratio Dynamic Time Warping (CDTW) is built upon Dynamic Time Warping (DTW) and is used to measure the similarity between two time series. In our preliminary work, we have explored optimizing Euclidean distance based on previous studies. However, Euclidean distance is more suitable for handling sequences of equal length and is prone to unnatural alignments. DTW addresses this issue by computing the accumulated distance between time series using dynamic programming, as shown in Eq. (1). D is a matrix and serves as a core data structure in the DTW computation process. The number of rows in this matrix corresponds to the length of the first time series X, and the number of columns corresponds to the length of the second time series Y. Each cell D(i, j) in the matrix stores an accumulated distance. The meaning of this accumulated distance D(i, j) is the optimal alignment between the first i points of the first sequence and the first j points of the second sequence. The DTW distance is a scalar value that represents the minimum accumulated distance required to optimally align the two complete sequences. Complexity estimation (CE) measures the fluctuation level of a sequence, and its calculation is given in Eq. (2), where n is the length of sequence, and $\sum_{i=1}^{n-1}(x_{i+1} - x_i)^2$ represents the sum of the squared differences between consecutive points in the sequence. CDTW extends DTW by incorporating a complexity ratio(CF) with the aim of capturing the fluctuation differences between time series, as defined in Eq. (3). CE(1) and CE(2) represent the complexity estimation values of the two specific sequences involved in the CDTW computation, calculated according to Eq. (2). CE reflects the fluctuation magnitude of a sequence, while CF is used to indicate the relative difference in fluctuation magnitude between the two sequences. The final CDTW formulation is presented in Eq. (4). To reduce computational complexity, we leverage the multi-resolution property of time series through recursive downsampling, generating shorter sequences. As shown in Eq. (5), the use of the Sakoe-Chiba Band constraint can limit the search path so that the distance is computed only when the difference between the current alignment positions *i* and *j* of the two sequences does not exceed *W*. Here, *W* is a window width parameter that restricts the path to a banded region within ±W around the diagonal, which is applied during the computation of matrix *D*. This improves the computational efficiency of *DTW* and further reduces the computational cost.

$$DTW = d(x_i, y_j) + min(D(i-1, j), D(i, j-1), D(i-1, j-1)), \tag{1}$$

$$CE(X) = \sqrt{\sum_{i=1}^{n-1}(x_{i+1} - x_i)^2}, \tag{2}$$

$$CF = \frac{max(CE1, CE2)}{min(CE1, CE2)}, \tag{3}$$

$$CDTW = DTW \times CF, \tag{4}$$

$$|i - j| \leq W. \tag{5}$$

3.2.4 HDFPM Model

HDFPM is built upon the Temporal Convolutional Network (TCN) and LightGBM models. TCN employs dilated causal convolutions, which preserve causality while expanding the receptive field and maintaining lower computational cost, 1-D dilated causal convolutions as shown in Eq. (6). In this equation, $Z_t^{(m)}$ represents the value of the output feature map at time step t and channel m (i.e., the convolution output); Cin denotes the number of input channels; K is the kernel size; d is the dilation factor, which

controls the step size of the convolution and $w_k^{(i,m)}$ represents the convolution kernel weight. Based on the TCN architecture, the model integrates extended 1D convolutions with max pooling layers, and incorporates an attention mechanism, as illustrated in Fig. 5. This structure increases the number of channels while reducing the temporal length of the input. The network stacks three convolutional blocks to capture temporal dependencies across time series. Each block consists of two dilated 1D convolutions with a kernel size of 3 × 1 and dilation rates of d = 2, 2, and 4, followed by ReLU activation and normalization. The number of channels in each block is set to 32, 64, and 128, respectively. Each block is followed by a max pooling layer that halves the time dimension. An attention mechanism is introduced before the final fully connected layer to enhance feature representation. After processing by the TCN module, the extracted features are passed to LightGBM for classification and model training.

$$Z_t^{(m)} = \sum_{i=1}^{Cin} \sum_{k=0}^{K-1} x_i(t - k \cdot d) w_k^{(i,m)}. \tag{6}$$

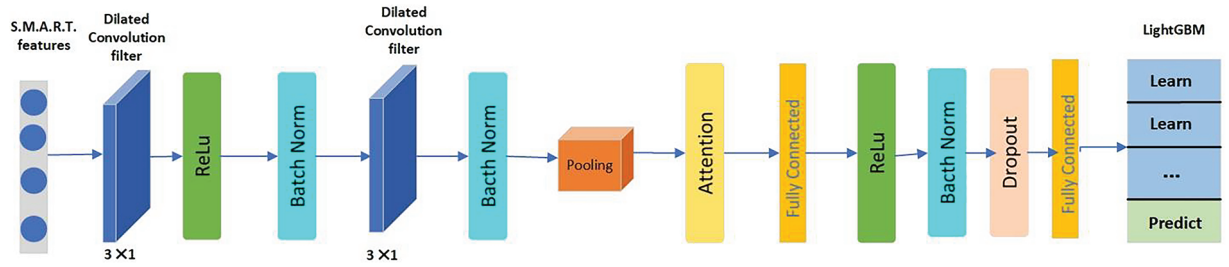


Figure 5: The structure of HDFPM

3.2.5 Impact of Shared Attributes across Heterogeneous Hard Drive Models on Modeling

In heterogeneous environments, the proportion of shared attributes between different HDD models can also affect the prediction performance of the model. In this paper, we define a shared attribute as a SMART attribute that exhibits an equal missing rate across two models. The proportion of shared attributes is calculated as shown in Eq. (7) where A represents the attribute set of HDD model A, and B represents the attribute set of another HDD model B. We selected several Seagate models—ST16000NM001G, ST12000NM0008, ST4000DM000, and ST14000NM001G—to compute their shared attribute ratios and evaluate their corresponding prediction performance. We argue that a higher proportion of shared attributes between two models generally leads to better prediction performance in the modeling process.

$$Shared\ Ratio = \frac{|A \cap B|}{|AUB|}. \tag{7}$$

4 Experiments and Results

In this chapter, we conduct experimental evaluations of the HDD failure prediction model using datasets with varying degrees of heterogeneity (Dataset-1, Dataset-2, Dataset-3, and Dataset-4). The experimental content includes:

Section 4.1 Introduction of the evaluation metrics;

Section 4.2 Performance comparison between the proposed HDFPM model and other baseline models, including CNN_LSTM, CNN_BiLSTM, XGBoost, GNN, and the latest HDD failure prediction model SiaDFP [11];

Section 4.3 Comparison of prediction performance on datasets processed with CDTW, DTW, and without either.

Section 4.4 Optimal Time Prediction Window;

Section 4.5 Impact of shared attributes across heterogeneous hard drive models on modeling;

Section 4.6 Evaluation of the robustness and statistical significance of the HDFPM model;

4.1 Evaluation Metrics

For evaluating HDD failure prediction, we use the TPR, FPR, AUC, F1-Score, and Optimal Time Prediction Window as the performance metrics.

The Confusion Matrix is an N × N table used to summarize the performance of a classification model. When N = 2, it represents a binary classification task. As shown in Table 2, this binary confusion matrix includes the following components: TP (True Positive): the number of failed hard drives correctly classified as failures; FP (False Positive): the number of healthy hard drives incorrectly classified as failures; TN (True Negative): the number of healthy hard drives correctly classified as healthy; FN (False Negative): the number of failed hard drives incorrectly classified as healthy.

Table 2: Binary confusion matrix

		Actual	
		Positive	**Negative**
Predicted	Postitive	TP	FP
	Negative	FN	TN

The True Positive Rate (TPR) is defined as the ratio of correctly predicted positive instances to the total number of actual positive instances. In general, a higher TPR in hard drive time-series classification models indicates a stronger ability to correctly identify failing drives. The formula is provided in Eq. (8).

$$TPR = \frac{TP}{TP + FN}. \tag{8}$$

The False Positive Rate (FPR) is defined as the ratio of incorrectly predicted positive instances to the total number of actual negative instances. In general, a lower FPR in hard drive time-series classification models indicates a lower probability of misclassifying healthy drives as failing. Under the condition of a high TPR, a lower FPR reflects better model performance. The formula is provided in Eq. (9).

$$FPR = \frac{FP}{FP + TN}. \tag{9}$$

The Receiver Operating Characteristic (ROC) Curve is a plot of the True Positive Rate (TPR) against the False Positive Rate (FPR) at various classification thresholds, where TPR is represented on the y-axis and FPR on the x-axis. The Area Under the ROC Curve (AUC) refers to the area under the ROC curve. A larger AUC value indicates better overall performance of the classification model.

The F1-Score represents the overall performance of a classification model. It considers both precision and recall in binary classification, and computes their harmonic mean to produce a balanced performance measure. The value of the F1-Score ranges from 0 to 1, with values closer to 1 indicating better overall model

performance. The formula is provided in Eq. (10).

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall}. \tag{10}$$

4.2 Experiments and Performance Evaluation

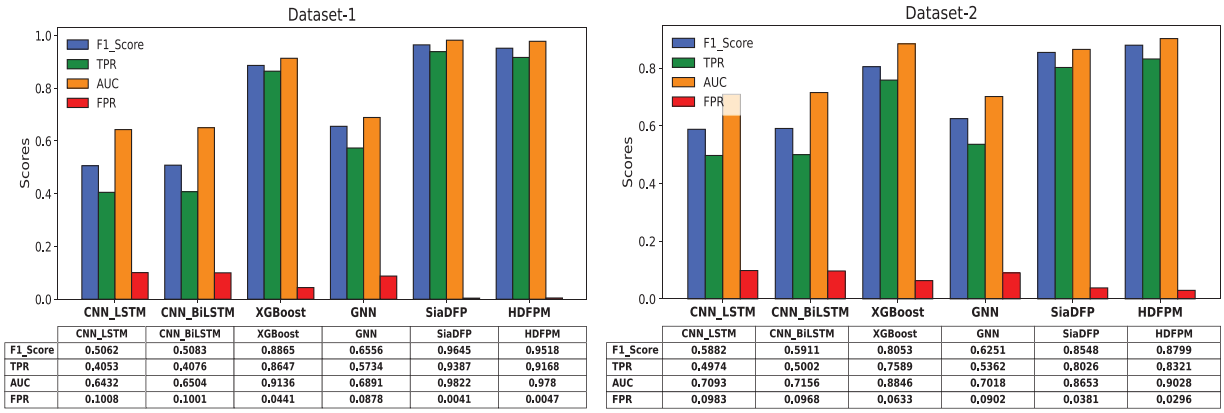
4.2.1 Prediction Performance of Various HDD Models under the Same Brand

The intra-brand model comparison experiments are divided into two parts. In the first part, two Seagate HDD models from Q1 2023—ST16000NM001G and ST12000NM0008—were selected to construct Dataset-1. The numbers of positive and negative samples are shown in Table 3. A total of 70% of the data was used for model training, while the remaining 30% was used as the testing set. To address the class imbalance issue, an oversampling technique was applied to balance the training dataset. The test set maintained the original data distribution to closely reflect real-world conditions.

Table 3: Comparison of healthy and failed disks in dataset-1

Disks	Count
Healthy	3,671,750
Failed	182

As shown in Fig. 6a, the proposed HDFPM model demonstrated significant performance improvements compared to the CNN_LSTM, XGBoost, and GNN models. However, on datasets with mild heterogeneity, its performance was slightly inferior to that of the SiaDFP model. The SiaDFP model achieved a TPR of 0.9387, an F1-Score of 0.9645, and an AUC of 0.9822 on this dataset.



	CNN_LSTM	CNN_BiLSTM	XGBoost	GNN	SiaDFP	HDFPM
F1_Score	0.5062	0.5083	0.8865	0.6556	0.9645	0.9518
TPR	0.4053	0.4076	0.8647	0.5734	0.9387	0.9168
AUC	0.6432	0.6504	0.9136	0.6891	0.9822	0.978
FPR	0.1008	0.1001	0.0441	0.0878	0.0041	0.0047

(a)

	CNN_LSTM	CNN_BiLSTM	XGBoost	GNN	SiaDFP	HDFPM
F1_Score	0.5882	0.5911	0.8053	0.6251	0.8548	0.8799
TPR	0.4974	0.5002	0.7589	0.5362	0.8026	0.8321
AUC	0.7093	0.7156	0.8846	0.7018	0.8653	0.9028
FPR	0.0983	0.0968	0.0633	0.0902	0.0381	0.0296

(b)

Figure 6: The testing results of CNN_LSTM, CNN_BiLSTM, XGBoost, GNN, SiaDFP and HDFPM models on Dataset-1 and Dataset-2 are presented. (**a**) shows the testing results of each model on Dataset-1; (**b**) presents the testing results on Dataset-2

The second part involves selecting 28 Seagate hard drive models from Q1 2023. The numbers of positive and negative samples are shown in Table 4. A total of 70% of the data was used as the training set and the remaining 30% as the testing set. To address the class imbalance, an oversampling technique was applied to the training set. The test set maintained the original data distribution to closely reflect real-world conditions.

Based on Dataset-2, we trained and evaluated the CNN_LSTM, CNN_BiLSTM, XGBoost, GNN, SiaDFP and the proposed HDFPM models.

Table 4: Comparison of healthy and failed disks in dataset-2

Disks	Count
Healthy	10,104,518
Failed	643

As shown in Fig. 6b, the experimental results demonstrate the prediction performance across 28 Seagate HDD models of the same brand. The CNN_BiLSTM model achieved a 0.28 improvement in TPR compared to the CNN_LSTM model. The proposed HDFPM model outperformed the LSTM-based models, XGBoost model, GNN model, and SiaDFP model. Among them, the HDFPM model achieved the best performance, with a TPR of 0.8321, an F1-Score of 0.8799, and an AUC of 0.9028.

4.2.2 Prediction Performance of Different Models across Brands

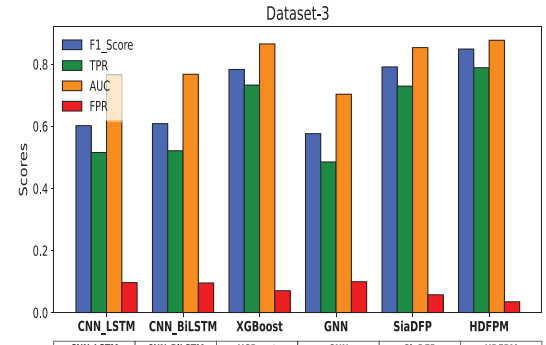
The brand and model comparison experiments are divided into two parts. In the first part, 41 HDD models from two brands, Seagate and Hitachi, in Q1 2023 were selected to construct Dataset-3. The numbers of positive and negative samples are shown in Table 5. A total of 70% of the data was used as the training set, and the remaining 30% as the testing set. To address the class imbalance, an oversampling method was applied to the training set.

Table 5: Comparison of healthy and failed disks in dataset-3

Disks	Count
Healthy	14,072,618
Failed	763

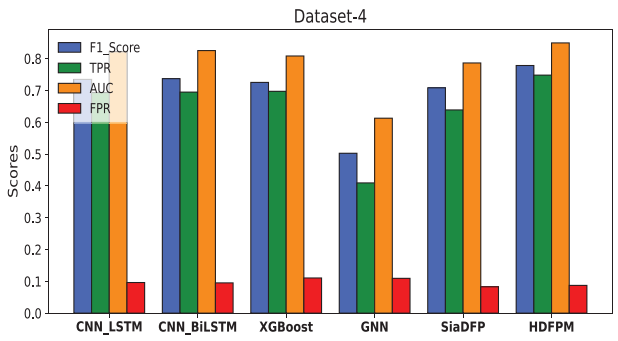
The experimental results are shown in Fig. 7a, which illustrates the prediction performance on Dataset-3. It can be observed that the CNN_BiLSTM model shows a slight improvement over the CNN_LSTM model, and there is a small overall improvement in prediction performance compared to Dataset-1 and Dataset-2. The GNN and SiaDFP models exhibit a significant performance decline compared to Dataset-1 and Dataset-2. The XGBoost and HDFPM models show a slight decrease in performance on Dataset-3 compared to Dataset-1 and Dataset-2. However, the HDFPM model still achieves the best performance, with a TPR of 0.7886, an F1-Score of 0.8487, and an AUC of 0.8769.

The second part involves selecting Dataset-4, which consists of 65 HDD models from over 10 different brands, with the numbers of positive and negative samples shown in Table 6. An oversampling method was also applied to the training set to balance the dataset. The testing results of the CNN_LSTM, CNN_BiLSTM, XGBoost, GNN, SiaDFP, and HDFPM models trained on Dataset-4 are presented.



	CNN_LSTM	CNN_BiLSTM	XGBoost	GNN	SiaDFP	HDFPM
F1_Score	0.602	0.6085	0.7829	0.5762	0.7911	0.8487
TPR	0.5154	0.5212	0.7327	0.4851	0.7292	0.7886
AUC	0.7653	0.7674	0.8651	0.7032	0.8533	0.8769
FPR	0.0966	0.0954	0.0705	0.0994	0.0574	0.0349

(a)



	CNN_LSTM	CNN_BiLSTM	XGBoost	GNN	SiaDFP	HDFPM
F1_Score	0,7345	0,7369	0,7252	0,5025	0,7082	0,7783
TPR	0,6929	0,6947	0,6972	0,4093	0,6387	0,7481
AUC	0,8214	0,8252	0,8082	0,6127	0,7863	0,8493
FPR	0,0962	0,0951	0,1105	0,1094	0,0828	0,0872

(b)

Figure 7: The testing results of CNN_LSTM, CNN_BiLSTM, XGBoost, GNN, SiaDFP and HDFPM models on Dataset-3 and Dataset-4 are presented. (**a**) shows the testing results of each model on Dataset-3; (**b**) presents the testing results on Dataset-4

Table 6: Comparison of healthy and failed disks in dataset-4

Disks	Count
Healthy	21,245,165
Failed	925

The experimental results are shown in Fig. 7b, which illustrates the prediction performance on Dataset-4. It can be seen that CNN_BiLSTM outperforms CNN_LSTM, and shows a significant improvement compared to Dataset-1. The GNN and SiaDFP models exhibit performance degradation compared to Dataset-1, Dataset-2, and Dataset-3. The performance of the XGBoost and HDFPM models also declines on Dataset-4 compared to Dataset-1, Dataset-2, and Dataset-3. However, the HDFPM model still achieves the best performance and demonstrates the highest stability, with a TPR of 0.7481, an F1-Score of 0.7783, and an AUC of 0.8493.

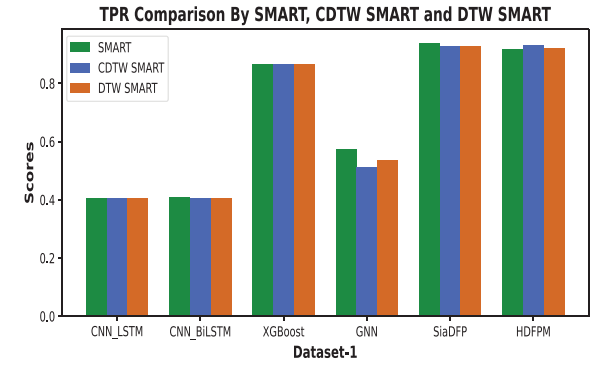
The reason that CNN_LSTM and CNN_BiLSTM perform better on heterogeneous datasets than in homogeneous datasets is that these models are deep learning models based on deep time-series data. The more continuously changing data in heterogeneous datasets allow CNN_BiLSTM to learn long-term dependencies, improving accuracy. On the other hand, XGBoost and LightGBM are ensemble learning models, which may lead to slight fluctuations in prediction accuracy. Based on this, the proposed HDFPM uses TCN to process features, making the prediction performance more stable. In summary, although the SiaDFP model slightly outperforms HDFPM on Dataset-1, HDFPM demonstrates superior and more stable performance in more complex heterogeneous environments. The poor performance of the GNN model is mainly attributed to the lack of topological relationships among samples in the dataset, as GNN models primarily focus on capturing interactions between nodes and their neighbors.

4.3 CDTW Comparison Experiment

To separately evaluate the impact of CDTW and DTW feature enhancements on model performance, we applied CDTW and DTW processing individually to the previous four datasets and added the corresponding

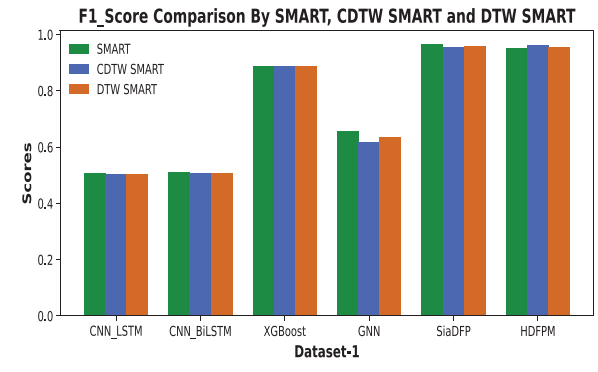
CDTW or DTW features to the original SMART attributes. These four datasets (Dataset-1, Dataset-2, Dataset-3, and Dataset-4) were individually enriched with CDTW features and DTW features. Subsequently, we trained the CNN_LSTM, CNN_BiLSTM, XGBoost, GNN, SiaDFP, and HDFPM models and conducted comparative performance analysis.

The TPR comparison results on Dataset-1 are shown in Fig. 8a. After adding CDTW-processed features, the TPR values of the CNN_LSTM, CNN_BiLSTM, and SiaDFP models showed a slight decrease, while the GNN model exhibited a more noticeable decline. In contrast, the TPR values of the XGBoost and HDFPM models showed a slight improvement. After adding DTW-processed features, the TPR values of the CNN_LSTM, CNN_BiLSTM, and SiaDFP models also experienced a slight decrease; however, the decline was more minor compared to CDTW. The GNN model still showed a significant decrease in TPR, although the negative impact of DTW on the GNN model was less severe than that of CDTW. The TPR values of the XGBoost and HDFPM models still exhibited slight improvements compared to the results without any feature processing, though the improvements were not as pronounced as those achieved with CDTW processing. The F1-Score comparison results are shown in Fig. 8b. After adding DTW- and CDTW-processed features, the F1-Score values of the CNN_LSTM, CNN_BiLSTM, and SiaDFP models all showed a slight decrease, while the GNN model experienced a more significant drop. The negative impact and positive improvement effects of CDTW on the models were both more pronounced.



	SMART	CDTW SMART	DTW SMART
CNN_LSTM	0.4053	0.4042	0.4047
CNN_BiLSTM	0.4076	0.4059	0.4066
XGBoost	0.8647	0.8658	0.8651
GNN	0.5734	0.5128	0.5346
SiaDFP	0.9387	0.928	0.9277
HDFPM	0.9168	0.9305	0.9213

(a)

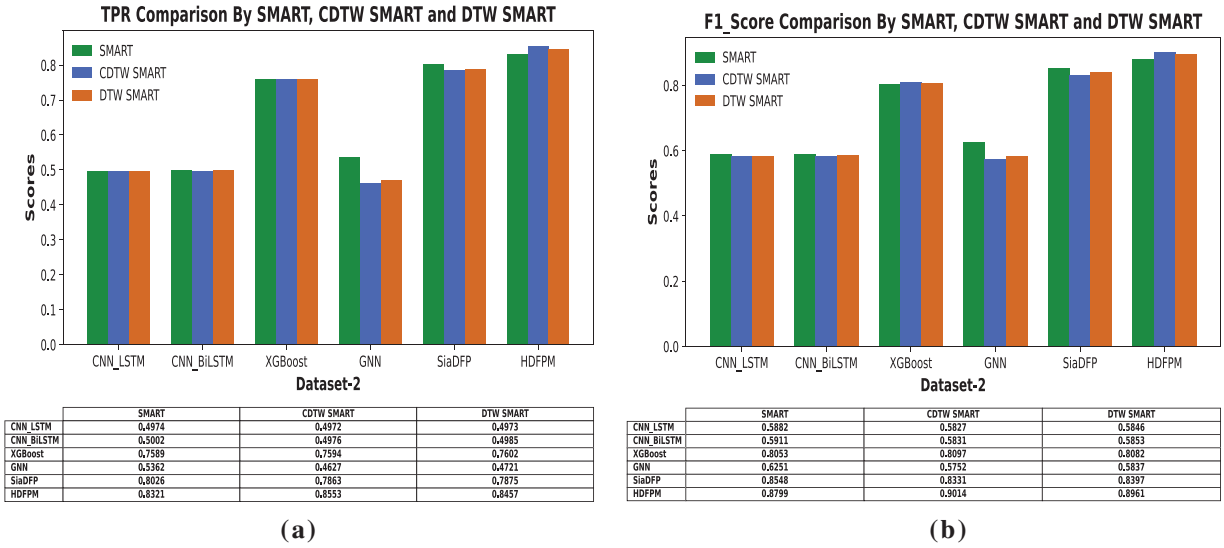


	SMART	CDTW SMART	DTW SMART
CNN_LSTM	0.5062	0.5036	0.5043
CNN_BiLSTM	0.5083	0.5054	0.5065
XGBoost	0.8865	0.8883	0.8869
GNN	0.6556	0.6175	0.6328
SiaDFP	0.9645	0.9542	0.9567
HDFPM	0.9518	0.9597	0.9546

(b)

Figure 8: The impact of CDTW-processed and DTW-processed feature enhancement on various models is shown. (**a**) represents the TPR testing results on Dataset-1, and (**b**) represents the F1-Score testing results on Dataset-1

The TPR and F1-Score comparison results on Dataset-2 are shown in Fig. 9. After adding CDTW-processed features, the TPR and F1-Score values of the CNN_LSTM, CNN_BiLSTM, and SiaDFP models all showed a slight decrease, while the GNN model still exhibited a noticeable decline. In contrast, the TPR and F1-Score values of the XGBoost and HDFPM models showed a slight improvement. After adding DTW-processed features, the TPR and F1-Score values of the CNN_LSTM, CNN_BiLSTM, and SiaDFP models also experienced a slight decrease, though the negative impact of DTW was relatively minor. The metrics of the GNN model still showed a significant decrease, although the negative impact of DTW on the GNN model was less severe than that of CDTW. The TPR and F1-Score values of the XGBoost and HDFPM models still showed slight improvements compared to those without any feature processing, but the improvements were not as pronounced as those achieved with CDTW processing.



	SMART	CDTW SMART	DTW SMART
CNN_LSTM	0.4974	0.4972	0.4973
CNN_BiLSTM	0.5002	0.4976	0.4985
XGBoost	0.7589	0.7594	0.7602
GNN	0.5362	0.4627	0.4721
SiaDFP	0.8026	0.7863	0.7875
HDFPM	0.8321	0.8553	0.8457

(a)

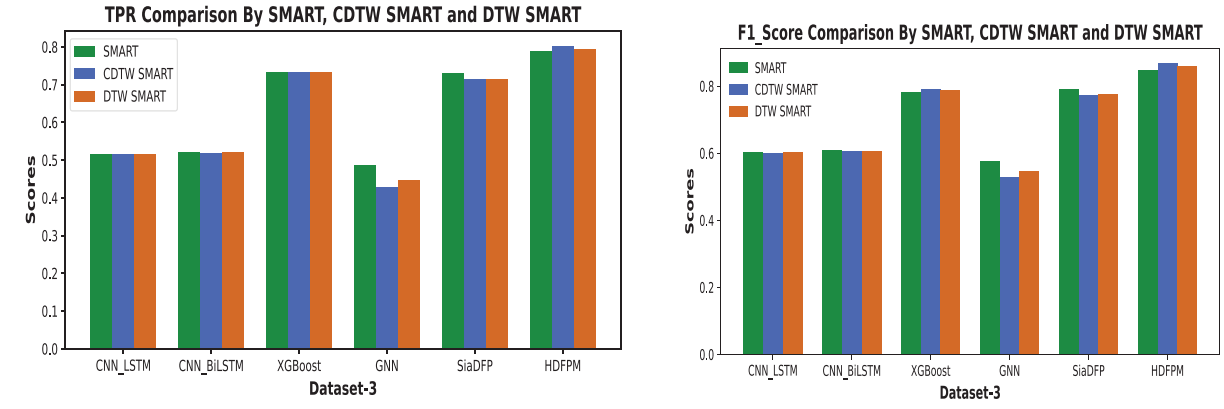
	SMART	CDTW SMART	DTW SMART
CNN_LSTM	0.5882	0.5827	0.5846
CNN_BiLSTM	0.5911	0.5831	0.5853
XGBoost	0.8053	0.8097	0.8082
GNN	0.6251	0.5752	0.5837
SiaDFP	0.8548	0.8331	0.8397
HDFPM	0.8799	0.9014	0.8961

(b)

Figure 9: The impact of CDTW-processed and DTW-processed feature enhancement on various models is shown. (**a**) represents the TPR testing results on Dataset-2, and (**b**) represents the F1-Score testing results on Dataset-2

The TPR comparison results on Dataset-3 are shown in Fig. 10a. After adding CDTW-processed features, the GNN model showed a noticeable decline in its metrics, while the TPR values of the CNN_LSTM, CNN_BiLSTM, and SiaDFP models exhibited only slight changes. In contrast, the TPR values of the XGBoost and HDFPM models showed a small improvement. After adding DTW-processed features, the TPR values of the CNN_LSTM, CNN_BiLSTM, SiaDFP, and GNN models also declined, but the decrease was smaller compared to CDTW processing. The TPR values of the XGBoost and HDFPM models still improved compared to those without any feature processing, though the improvement was less significant than that achieved with CDTW processing. The F1-Score comparison results are shown in Fig. 10b. CDTW yielded more significant improvements for XGBoost and HDFPM compared to DTW; however, its negative impact on CNN_LSTM, CNN_BiLSTM, SiaDFP, and GNN was also more pronounced than that of DTW.

The TPR and F1-Score comparison results on Dataset-4 are shown in Fig. 11. After adding CDTW-processed and DTW-processed features, the results of CNN_LSTM and CNN_BiLSTM remain nearly unchanged, while the TPR and F1-Score values of the XGBoost and HDFPM models show a slight improvement, with the impact of CDTW being more pronounced. For the negative effect on the GNN model, CDTW is also more significant.



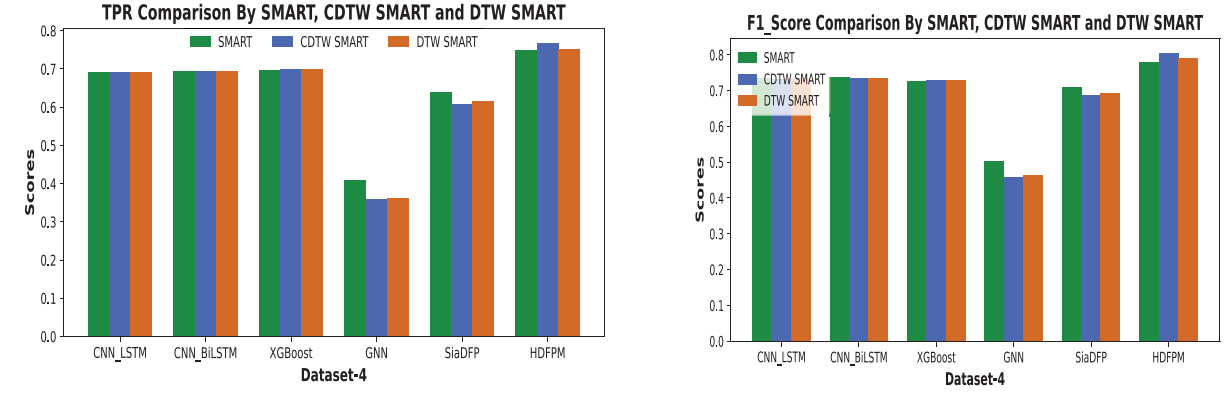
	SMART	CDTW SMART	DTW SMART
CNN_LSTM	0.5154	0.5153	0.5152
CNN_BiLSTM	0.5212	0.5193	0.5198
XGBoost	0.7327	0.7332	0.7329
GNN	0.4851	0.4278	0.4463
SiaDFP	0.7292	0.7129	0.7138
HDFPM	0.7886	0.8013	0.7944

(a)

	SMART	CDTW SMART	DTW SMART
CNN_LSTM	0.602	0.6007	0.6013
CNN_BiLSTM	0.6085	0.6052	0.6069
XGBoost	0.7829	0.7918	0.7863
GNN	0.5762	0.5396	0.5463
SiaDFP	0.7911	0.7736	0.7745
HDFPM	0.8487	0.8683	0.8605

(b)

Figure 10: The impact of CDTW-processed and DTW-processed feature enhancement on various models is shown. (**a**) represents the TPR testing results on Dataset-3, and (**b**) represents the F1-Score testing results on Dataset-3



	SMART	CDTW SMART	DTW SMART
CNN_LSTM	0.6929	0.6921	0.6923
CNN_BiLSTM	0.6947	0.6934	0.6938
XGBoost	0.6972	0.7008	0.6995
GNN	0.4093	0.3586	0.3633
SiaDFP	0.6387	0.6083	0.6172
HDFPM	0.7481	0.7675	0.7532

(a)

	SMART	CDTW SMART	DTW SMART
CNN_LSTM	0.7345	0.7322	0.7334
CNN_BiLSTM	0.7369	0.7338	0.7351
XGBoost	0.7252	0.7296	0.7283
GNN	0.5025	0.4578	0.463
SiaDFP	0.7082	0.6874	0.6936
HDFPM	0.7783	0.8051	0.7914

(b)

Figure 11: The impact of CDTW-processed and DTW-processed feature enhancement on various models is shown. (**a**) represents the TPR testing results on Dataset-4, and (**b**) represents the F1-Score testing results on Dataset-4

CDTW features are model-dependent, providing slight improvements for XGBoost and HDFPM, while having little to no effect or even a negative impact on other models. CNN_LSTM and CNN_BiLSTM are time-series-based models, so CDTW-processed features have minimal influence on their performance. In contrast, XGBoost and HDFPM are decision tree-based models, and CDTW feature enhancement helps them better leverage time-series features. The SiaDFP model has already extracted time-series features through its integrated 2D-Attention, CP-MAP, and LSTM modules, making CDTW and DTW features redundant and causing performance interference. The GNN model is not well-suited for handling high-dimensional features, leading to degraded prediction performance.

4.4 Optimal Time Prediction Window

To determine the optimal prediction time window for each model, we selected the last 10 days before the failure date of each failed sample in Dataset-2 and Dataset-4 as the analysis window. The labels were set to 1 in sequence for each of the 10 days preceding the actual failure. Based on this setup, multiple experiments were conducted to observe the models' effectiveness in advance failure prediction. As shown in Fig. 12, the best performance was achieved when predicting one day in advance. Moreover, although the GNN and SiaDFP models exhibited some oscillations, the F1-scores of all models generally decreased as the prediction window moved further from the actual failure date, with a more pronounced downward trend as the distance increased. The proposed HDFPM model achieved the best performance for one-day-ahead failure prediction and demonstrated stable performance even for three-day-ahead predictions. Considering that backing up and migrating data from hard drives on the verge of failure typically requires several hours, and taking into account cases involving data fragmentation, backups can generally be completed within several hours. Therefore, the proposed HDFPM model enables timely prediction of hard drive failures, allowing data centers to migrate and back up data in a timely manner.

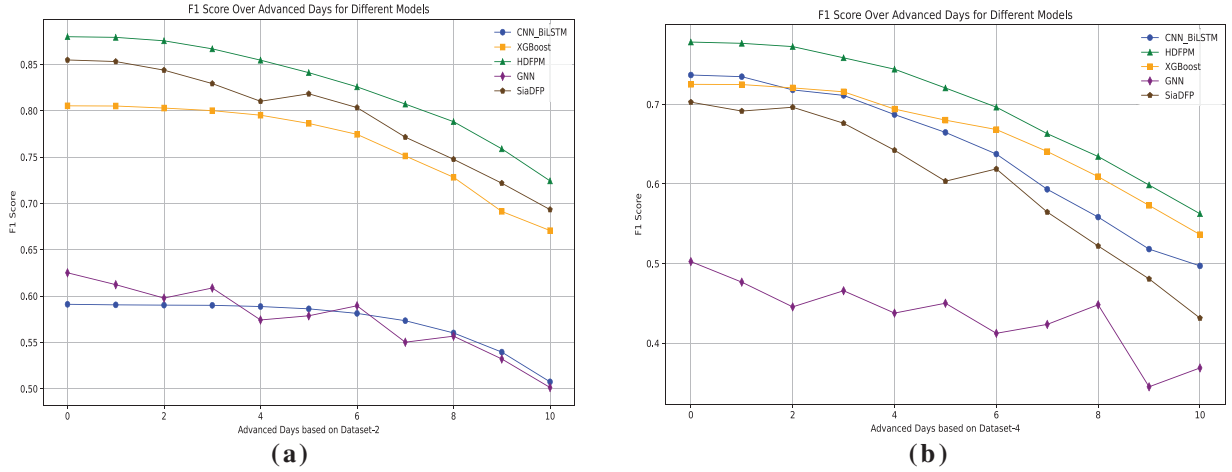


Figure 12: Advanced prediction capability of different models measured by F1-Score. (**a**) Advanced days based on Dataset-2 results. (**b**) Advanced days based on Dataset-4 results

4.5 Impact of Shared Attributes across Heterogeneous Hard Drive Models on Modeling

The number of disk blocks for Seagate models ST16000NM001G, ST12000NM0008, ST16000NM002J, and ST500LM021 is listed in Table 7. The shared attribute ratios between ST16000NM001G and the other models are presented in Table 8. Dataset-1 corresponds to the previously mentioned dataset composed of ST16000NM001G and ST12000NM0008. Dataset-5 consists of ST16000NM001G and ST16000NM002J, while Dataset-6 consists of ST16000NM001G and ST500LM021.

Table 7: The number of hard drives for four different models of the ST brand

Model	Count
ST16000NM001G	1,897,163
ST12000NM0008	1,774,769
ST16000NM002J	27,590
ST500LM021	2738

Table 8: Common attribute ratios—ST16000NM001G and other models

Model	Common_Attributes Ratio
ST12000NM0008	96.39%
ST16000NM002J	73.49%
ST500LM021	68.67%

For each dataset, features were selected from the set of shared attributes using Variance Inflation Factor (VIF) analysis and Recursive Feature Elimination (RFE). The prediction performance obtained using these selected features is reported in Table 9.

Table 9: The prediction performance of each model on datasets (Dataset-1, Dataset-5, and Dataset-6) with different ratios of common attributes

Dataset	XGBoost		CNN_BiLSTM		GNN		SiaDFP		HDFPM	
	F1-Score	TPR	F1-Score	TPR	F1-Score	TPR	F1-Score	TPR	F1-Score	TPR
Dataset-1	0.8865	0.8647	0.5083	0.4076	0.6556	0.5734	0.9645	0.9387	0.9518	0.9168
Dataset-5	0.8583	0.8411	0.4823	0.3851	0.6223	0.5394	0.9463	0.9233	0.9383	0.9035
Dataset-6	0.8471	0.8392	0.4757	0.3839	0.6089	0.5215	0.9449	0.9195	0.9364	0.8976

As shown in Table 9, among different hard drive models under the same brand, a higher proportion of shared attributes generally leads to better prediction performance.

4.6 Evaluation of the Robustness and Statistical Significance of the HDFPM Model

4.6.1 F1-Score Evaluation of the Robustness and Statistical Significance of the HDFPM Model

To assess the statistical robustness of the results and compare the significance between models, we employed a 10-fold cross-validation strategy on the Dataset-4 to evaluate the F1-Score performance of the HDFPM, CNN_BiLSTM, XGBoost, and SiaDFP models. Standard deviation and Wilcoxon p-value were used as evaluation metrics. The original dataset was randomly divided into ten folds. In each iteration, one fold was used as the test set, while the remaining nine folds were combined as the training set for model training. This experiment was repeated twice. The test results are shown in Tables 10 and 11.

Table 12 presents the robustness analysis of the first-round experiments. The results indicate that the HDFPM model achieves a higher average F1-Score and a lower standard deviation.

Table 13 presents the significance analysis of the first-round experiments. HDFPM demonstrates significantly better performance than the other models.

Tables 14 and 15 present the robustness and significance analysis of the second-round experiments for the HDFPM model and other models. The results demonstrate that the HDFPM model achieves a higher average F1-Score and a lower standard deviation. The HDFPM model also shows better statistical significance compared to the other models.

Table 10: F1-Scores of sampling subdatasets in the first round of 10-fold cross-validation

Sampling Subdataset				
F1_Score	**HDFPM**	**CNN_BiLSTM**	**XGBoost**	**SiaDFP**
Sampling subdataset-1	0.7742	0.7384	0.7249	0.7053
Sampling subdataset-2	0.7559	0.7587	0.7446	0.6974
Sampling subdataset-3	0.7553	0.7215	0.7302	0.7132
Sampling subdataset-4	0.7686	0.7168	0.7236	0.7228
Sampling subdataset-5	0.7734	0.7422	0.7291	0.7037
Sampling subdataset-6	0.7810	0.7265	0.7153	0.6946
Sampling subdataset-7	0.7537	0.7356	0.7542	0.7085
Sampling subdataset-8	0.7616	0.7482	0.7225	0.7263
Sampling subdataset-9	0.7524	0.7531	0.7478	0.7104
Sampling subdataset-10	0.7795	0.7310	0.7397	0.7316

Table 11: F1-Scores of sampling subdatasets in the second round of 10-fold cross-validation

Sampling Subdataset				
F1_Score	**HDFPM**	**CNN_BiLSTM**	**XGBoost**	**SiaDFP**
Sampling subdataset-1	0.7703	0.7321	0.7228	0.6997
Sampling subdataset-2	0.7534	0.7542	0.7546	0.7112
Sampling subdataset-3	0.7529	0.7439	0.7314	0.7309
Sampling subdataset-4	0.7694	0.7264	0.7453	0.7157
Sampling subdataset-5	0.7822	0.7183	0.7417	0.7188
Sampling subdataset-6	0.7527	0.7539	0.7368	0.7045
Sampling subdataset-7	0.7752	0.7350	0.7294	0.6903
Sampling subdataset-8	0.7551	0.7226	0.7139	0.7231
Sampling subdataset-9	0.7618	0.7258	0.7356	0.7129
Sampling subdataset-10	0.7507	0.7522	0.7379	0.7135

Table 12: Robustness analysis of each model's F1-Score in the first round of 10-fold cross-validation

Model	**Mean**	**Std**
HDFPM	0.7656	0.0109
CNN_BiLSTM	0.7372	0.0137
XGBoost	0.7332	0.0126
SiaDFP	0.7114	0.0122

Table 13: Significance analysis of each model's F1-Score in the first round of 10-fold cross-validation

Comparison models	*p*(wil)	Significance assessment
HDFPM and CNN_BiLSTM	0.0098	Significant
HDFPM and XGBoost	0.0039	Significant
HDFPM and SiaDFP	0.0020	significant

Table 14: Robustness analysis of each model's F1-Score in the second round of 10-fold cross-validation

Model	Mean	Std
HDFPM	0.7624	0.0112
CNN_BiLSTM	0.7364	0.0135
XGBoost	0.7349	0.0114
SiaDFP	0.7120	0.0116

Table 15: Significance analysis of each model's F1-Score in the second round of 10-fold cross-validation

Comparison models	*p*(wil)	Significance assessment
HDFPM and CNN_BiLSTM	0.0273	Significant
HDFPM and XGBoost	0.0039	Significant
HDFPM and SiaDFP	0.0020	Significant

4.6.2 TPR Evaluation of the Robustness and Statistical Significance of the HDFPM Model

To more thoroughly assess the statistical robustness of the results and compare the significance between models, we supplemented the evaluation with a 10-fold cross-validation strategy on Dataset-4 to assess the TPR performance of the HDFPM, CNN_BiLSTM, XGBoost, and SiaDFP models. Standard deviation and Wilcoxon *p*-value were used as evaluation metrics. The original dataset was randomly divided into ten folds. In each iteration, one fold was used as the test set, while the remaining nine folds were combined as the training set for model training. The experiment was repeated twice. The test results are shown in Tables 16 and 17.

Table 18 presents the robustness analysis of the TPR in first-round experiments. The results indicate that the HDFPM model achieves a higher average TPR and a lower standard deviation.

Table 19 presents the significance analysis of the TPR in first-round experiments. HDFPM demonstrates significantly better performance than the other models.

Tables 20 and 21 present the robustness and significance analysis of TPR for the HDFPM model and other models in the second-round experiments. The results demonstrate that the HDFPM model achieves a higher average TPR and a lower standard deviation. The HDFPM model also shows better statistical significance compared to the other models.

Table 16: TPR of sampling subdatasets in the first round of 10-fold cross-validation

Sampling Subdataset				
TPR	**HDFPM**	**CNN_BiLSTM**	**XGBoost**	**SiaDFP**
Sampling subdataset-1	0.7423	0.6977	0.6871	0.6405
Sampling subdataset-2	0.7216	0.7312	0.7229	0.6278
Sampling subdataset-3	0.7209	0.7237	0.7019	0.6573
Sampling subdataset-4	0.7378	0.6682	0.6930	0.6686
Sampling subdataset-5	0.7431	0.7034	0.6913	0.6421
Sampling subdataset-6	0.7482	0.6854	0.6785	0.6295
Sampling subdataset-7	0.7234	0.6981	0.7192	0.6387
Sampling subdataset-8	0.7320	0.7043	0.7238	0.6617
Sampling subdataset-9	0.7228	0.7232	0.7164	0.6609
Sampling subdataset-10	0.7479	0.6956	0.6942	0.6778

Table 17: TPR of sampling subdatasets in the second round of 10-fold cross-validation

Sampling Subdataset				
TPR	**HDFPM**	**CNN_BiLSTM**	**XGBoost**	**SiaDFP**
Sampling subdataset-1	0.7392	0.6913	0.6883	0.6432
Sampling subdataset-2	0.7243	0.7155	0.7136	0.6612
Sampling subdataset-3	0.7246	0.7108	0.6978	0.6792
Sampling subdataset-4	0.7405	0.6876	0.7102	0.6586
Sampling subdataset-5	0.7493	0.7195	0.7048	0.6574
Sampling subdataset-6	0.7169	0.7223	0.7013	0.6421
Sampling subdataset-7	0.7374	0.6962	0.6931	0.6310
Sampling subdataset-8	0.7270	0.6851	0.7255	0.6707
Sampling subdataset-9	0.7302	0.6836	0.6927	0.6563
Sampling subdataset-10	0.7158	0.7184	0.6899	0.6624

Table 18: Robustness analysis of each model's TPR in the first round of 10-fold cross-validation

Model	**Mean**	**Std.**
HDFPM	0.7331	0.0106
CNN_BiLSTM	0.7002	0.0189
XGBoost	0.7019	0.0163
SiaDFP	0.6504	0.0171

Table 19: Significance analysis of each model's TPR in the first round of 10-fold cross-validation

Comparison models	*p*(wil)	Significance assessment
HDFPM and CNN_BiLSTM	0.0273	significant
HDFPM and XGBoost	0.0039	significant
HDFPM and SiaDFP	0.0020	significant

Table 20: Robustness analysis of each model's TPR in the second round of 10-fold cross-validation

Model	Mean	Std.
HDFPM	0.7305	0.0110
CNN_BiLSTM	0.6991	0.0157
XGBoost	0.7008	0.0138
SiaDFP	0.6562	0.0153

Table 21: Significance analysis of each model's TPR in the second round of 10-fold cross-validation

Comparison models	*p*(wil)	Significance assessment
HDFPM and CNN_BiLSTM	0.0098	Significant
HDFPM and XGBoost	0.0020	Significant
HDFPM and SiaDFP	0.0020	Significant

5 Conclusion

In this paper, we propose a time-series-based hard drive failure prediction method, HDFPM, designed for heterogeneous environments. The method leverages TCN to process the selected features and combines them with a LightGBM model. In addition, by enhancing the time-series features of the dataset through CDTW, both TPR and F1-Score still achieve slight improvements, and this method exhibits specific model dependency. The proposed HDFPM model achieves an F1-Score of 0.9518 under the condition that the false positive rate is reduced to within 1% for different models under the same brand. For different brands and models, the F1-Score reaches 0.8487 with a false positive rate controlled within 3.5%. Additionally, for various Seagate models, the model achieves an F1-Score of 0.8792 for one-day-ahead failure prediction, which can assist data centers in timely migrating and backing up data from hard drives that are at risk of failure. The experimental results also indicate that within the same brand, a higher proportion of shared attributes among different models leads to better modeling and prediction performance. Compared to other models, the proposed method also demonstrates superior robustness and significantly better prediction effectiveness. In the future, we will continue to investigate the reasons behind the performance degradation of HDFPM in heterogeneous hard drive failure prediction scenarios and work on enhancing the model's generalization capability. At the same time, we will continue to improve CDTW to enhance feature representation and performance, as well as explore better approaches for enhancing time-series features and more effective strategies to address the overfitting problem. At the same time, we will also conduct research and optimization on cost-sensitive learning and Transformer models in the field of hard drive failure prediction.

Acknowledgement: The authors sincerely thank all the teachers for their constructive suggestions and also express gratitude to Roycom Information Technology Corporation for providing the data for this research.

Funding Statement: This research was supported by the Tianjin Manufacturing High Quality Development Special Foundation (No. 20232185), and the Roycom Foundation (No. 70306901).

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Zhongrui Jing, Hongzhang Yang; feasibility analysis: Zhongrui Jing, Hongzhang Yang; methodology: Zhongrui Jing, Hongzhang Yang; system development and testing: Zhongrui Jing; dataset provision: Jiangpu Guo; writing—original draft: Zhongrui Jing; writing—review and editing: Hongzhang Yang; funding acquisition: Hongzhang Yang. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Not applicable.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Xu S, Xu X. ConvTrans-TPS: a convolutional transformer model for disk failure prediction in large-scale network storage systems. In: 2023 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD); 2023 May 24–26; Rio de Janeiro, Brazil. IEEE; 2023. p. 1318–23. doi:10.1109/CSCWD57460.2023.10152728.
2. Shen J, Ren Y, Wan J, Lan Y. Hard disk drive failure prediction for mobile edge computing based on an LSTM recurrent neural network. Mob Inf Syst. 2021;2021:8878364. doi:10.1155/2021/8878364.
3. Coursey A, Nath G, Prabhu S, Sengupta S. Remaining useful life estimation of hard disk drives using bidirectional LSTM networks. In: 2021 IEEE International Conference on Big Data (Big Data); 2021 Dec 15–18; Orlando, FL, USA. IEEE; 2021. p. 4832–41. doi:10.1109/bigdata52589.2021.9671605.
4. Ahmed J, Green RC II. Cost aware LSTM model for predicting hard disk drive failures based on extremely imbalanced S.M.A.R.T. sensors data. Eng Appl Artif Intell. 2024;127:107339. doi:10.1016/j.engappai.2023.107339.
5. Lu S, Luo B, Patel T, Yao Y, Tiwari D, Shi W. Making disk failure predictions SMARTer! In: Proceedings of the 18th USENIX Conference on File and Storage Technologies (FAST 2020); 2020 Feb 24–27; Santa Clara, CA, USA. p. 151–67. doi:10.5555/3386691.3386706.
6. Fang X, Guan W, Li J, Cao C, Xia B. SiaDFP: a disk failure prediction framework based on Siamese neural network in large-scale data center. IEEE Trans Serv Comput. 2024;17(5):2890–903. doi:10.1109/TSC.2024.3394692.
7. Shen J, Wan J, Lim SJ, Yu L. Random-forest-based failure prediction for hard disk drives. Int J Distrib Sens Netw. 2018;14(11):155014771880648. doi:10.1177/1550147718806480.
8. Huang S, Fu S, Zhang Q, Shi W. Characterizing disk failures with quantified disk degradation signatures: an early experience. In: 2015 IEEE International Symposium on Workload Characterization; 2015 Oct 4–6; Atlanta, GA, USA. IEEE; 2015. p. 150–9. doi:10.1109/IISWC.2015.26.
9. Chakraborttii C, Litz H. Improving the accuracy, adaptability, and interpretability of SSD failure prediction models. In: Proceedings of the 11th ACM Symposium on Cloud Computing; 2020 Oct 19–21; New York, NY, USA: Association for Computing Machinery (ACM). p. 120–33. doi:10.1145/3419111.3421300.
10. Shirke SA, Jayakumar N, Patil S. Design and performance analysis of modern computational storage devices: a systematic review. Expert Syst Appl. 2024;250:123570. doi:10.1016/j.eswa.2024.123570.
11. Han S, Lee PPC, Shen Z, He C, Liu Y, Huang T. Toward adaptive disk failure prediction via stream mining. In: 2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS); 2020 Nov 29–Dec 1; Singapore: IEEE; 2020. p. 628–38. doi:10.1109/icdcs47774.2020.00044.

12. Botezatu MM, 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; 2016 Aug 13–17; San Francisco, CA, USA. p. 39–48. doi:10.1145/2939672.2939699.
13. Li Q, Li H, Zhang K. Prediction of HDD failures by ensemble learning. In: 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS); 2019 Oct 18–20; Beijing, China. IEEE; 2019. p. 237–40. doi:10.1109/icsess47205.2019.9040739.
14. Zhang XY, Feng D, Tan ZP, Xie YW, Zhao SF, Wei YY. LWCM: a lookahead-window constrained model for disk failure prediction in large data centers. J Comput Sci Technol. 2025;40(3):748–65. doi:10.1007/s11390-025-3850-4.
15. Züfle M, Krupitzer C, Erhard F, Grohmann J, Kounev S. To fail or not to fail: predicting hard disk drive failure time windows. In: Measurement, modelling and evaluation of computing systems. Cham, Switzerland: Springer International Publishing; 2020. p. 19–36. doi:10.1007/978-3-030-43024-5_2.
16. Kaur K, Kaur K. Failure prediction, lead time estimation and health degree assessment for hard disk drives using voting based decision trees. Comput Mater Contin. 2019;60(3):913–46. doi:10.32604/cmc.2019.07675.
17. Burrello A, Pagliari DJ, Bartolini A, Benini L, Macii E, Poncino M. Predicting hard disk failures in data centers using temporal convolutional neural networks. In: Euro-Par 2020: Parallel Processing Workshops. Cham, Switzerland: Springer International Publishing; 2021. p. 277–89. doi:10.1007/978-3-030-71593-9_22.
18. Wang G, Wang Y, Sun X. Multi-instance deep learning based on attention mechanism for failure prediction of unlabeled hard disk drives. IEEE Trans Instrum Meas. 2021;70:3513509. doi:10.1109/TIM.2021.3068180.
19. Sun X, Chakrabarty K, Huang R, Chen Y, Zhao B, Cao H, et al. System-level hardware failure prediction using deep learning. In: Proceedings of the 56th Annual Design Automation Conference 2019; 2019 Jun 2–6; Las Vegas, NV, USA. p. 1–6. doi:10.1145/3316781.3317918.
20. Zhang J, Wang J, He L, Li Z, Yu PS. Layerwise perturbation-based adversarial training for hard drive health degree prediction. In: 2018 IEEE International Conference on Data Mining (ICDM); 2018 Nov 17–20; Singapore. IEEE; 2018. p. 1428–33. doi:10.1109/ICDM.2018.00197.
21. Galicia A, Talavera-Llames R, Troncoso A, Koprinska I, Martínez-Álvarez F. Multi-step forecasting for big data time series based on ensemble learning. Knowl Based Syst. 2019;163:830–41. doi:10.1016/j.knosys.2018.10.009.
22. Sun S, Wei Y, Wang S. AdaBoost-LSTM ensemble learning for financial time series forecasting. In: Proceedings of the 18th International Conference on Computational Science (ICCS 2018); 2018 Jun 11–13; Wuxi, China. p. 590–7. doi:10.1007/978-3-319-93713-7_55.
23. Yang H, Li Z, Qiang H, Li Z, Tu Y, Yang Y. ZTE-predictor: disk failure prediction system based on LSTM. In: 2020 50th Annual IEEE-IFIP International Conference on Dependable Systems and Networks-Supplemental Volume (DSN-S); 2020 Jun 29–Jul 2; Valencia, Spain. IEEE; 2020. p. 17–20. doi:10.1109/dsn-s50200.2020.00017.
24. Gao G, Wu P, Li H, Zhang T. Disk failure prediction based on transfer learning. In: Proceedings of the 18th International Conference on Intelligent Computing (ICIC 2022); 2022 Aug 7–11; Xi'an, China. p. 628–37. doi:10.1007/978-3-031-13829-4_54.
25. Wang Y, He L, Jiang S, Chow TWS. Failure prediction of hard disk drives based on adaptive Rao-blackwellized particle filter error tracking method. IEEE Trans Ind Inf. 2021;17(2):913–21. doi:10.1109/tii.2020.3016121.
26. Zhou Y, Wang F, Feng D. A disk failure prediction method based on active semi-supervised learning. ACM Trans Storage. 2022;18(4):1–33. doi:10.1145/3523699.
27. Lima FDS, Pereira FLF, Chaves IC, Machado JC, Gomes JPP. Predicting the health degree of hard disk drives with asymmetric and ordinal deep neural models. IEEE Trans Comput. 2021;70(2):188–98. doi:10.1109/tc.2020.2987018.
28. Zhang Y, Hao W, Niu B, Liu K, Wang S, Liu N, et al. Multi-view feature-based SSD failure prediction: what, when, and why. In: Proceedings of the 21st USENIX Conference on File and Storage Technologies (FAST 2023); 2023 Feb 20–23; Santa Clara, CA, USA. p. 409–24.