

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



PHAN VĂN PHÙNG

**XÂY DỰNG HỆ THỐNG PHÁT HIỆN LỖI Ổ CỨNG
TRONG MẠNG MÁY TÍNH QUÂN SỰ
CỦA BINH CHỦNG THÔNG TIN LIÊN LẠC**

TÓM TẮT ĐỀ ÁN THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

HÀ NỘI - 2025

Luận văn được hoàn thành tại:
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Người hướng dẫn khoa học: PGS. TSKH. HOÀNG ĐĂNG HẢI

Phản biện 1:

Phản biện 2:

Luận văn sẽ được bảo vệ trước Hội đồng chấm luận văn thạc sĩ tại Học viện
Công nghệ Bưu chính Viễn thông

Vào lúc: giờ ngày tháng năm

Có thể tìm hiểu luận văn tại:

- Thư viện của Học viện Công nghệ Bưu chính Viễn thông

MỞ ĐẦU

Mạng máy tính quân sự của Binh chủng là mạng độc lập, không liên thông kết nối với mạng Internet và các hệ thống thông tin dân sự khác, được triển khai rộng khắp với hơn 2600 máy tính kết nối, 01 trung tâm dữ liệu, 10 phòng máy chủ (gần 100 máy chủ vật lý). Do đặc điểm đòi hỏi độ bảo mật thông tin cao, hệ thống mạng hoàn toàn độc lập nên việc sử dụng dịch vụ lưu trữ của bên thứ ba như Google cloud, iCloud, One drive, ... là điều không thể thực hiện được. Dữ liệu của người dùng chủ yếu được lưu trữ trên các máy tính cục bộ nên vẫn tồn tại khá nhiều nguy cơ hư hỏng các ổ cứng trên máy chủ, gây mất mát dữ liệu, làm gián đoạn thông tin liên lạc, tốn nhiều thời gian và công sức để phát hiện lỗi và khôi phục hệ thống. Việc theo dõi giám sát tình trạng hoạt động, phát hiện và dự đoán lỗi ổ cứng trong hệ thống mạng của Binh chủng đang là một nhu cầu cấp thiết.

Việc xây dựng hệ thống có khả năng đưa ra cảnh báo và dự báo hỏng hóc không chỉ có ý nghĩa trong việc bảo vệ dữ liệu người dùng, giảm thời gian downtime của hệ thống mà còn có ý nghĩa trong việc xây dựng các kế hoạch bảo dưỡng các trang thiết bị trong đơn vị một cách chính xác, hiệu quả và tiết kiệm, khắc phục được một số hạn chế trong công tác bảo dưỡng các trang thiết bị công nghệ thông tin hiện nay. Dự báo được tình trạng hỏng hóc còn hỗ trợ người quản lý, chỉ huy xây dựng kế hoạch, ra quyết định mua sắm các trang thiết bị dự phòng, thay thế, vật tư bảo đảm kỹ thuật một cách khoa học, hiệu quả.

Từ những lý do trên em đã lựa chọn đề tài "***Xây dựng hệ thống phát hiện lỗi ổ cứng trong mạng máy tính quân sự của Binh chủng Thông tin liên lạc***".

Nội dung đề án được chia thành 3 chương như sau:

- CHƯƠNG 1. Tổng quan về hệ thống phát hiện lỗi ổ cứng: Trình bày các cơ sở lý thuyết và thực tiễn liên quan đến bài toán giám sát, phát hiện và dự báo lỗi đĩa cứng trong các mạng chuyên dụng như mạng quân sự của Binh chủng Thông tin liên lạc, Bộ Quốc phòng.

- CHƯƠNG 2. Nghiên cứu xây dựng mô hình học máy cho phát hiện lỗi ổ cứng: Nghiên cứu xây dựng mô hình kiến trúc hệ thống và các thành phần hệ thống; nghiên cứu phương pháp phân tích, đánh giá đặc điểm dữ liệu SMART của ổ cứng; lựa chọn mô hình học máy phù hợp.

- CHƯƠNG 3. Triển khai thử nghiệm và đánh giá kết quả: Thu thập bộ dữ liệu lỗi ổ cứng tại đơn vị và thực hiện gán nhãn để huấn luyện. Triển khai cài đặt mô hình học máy dự báo hỏng hóc ổ cứng để thực nghiệm. Triển khai huấn luyện mô hình, đánh giá độ chính xác mô hình với tập dữ liệu huấn luyện đã thu thập.

Chương 1: TỔNG QUAN

VỀ HỆ THỐNG PHÁT HIỆN LỖI Ổ CỨNG

Chương này trình bày các cơ sở lý thuyết và thực tiễn liên quan đến bài toán giám sát, phát hiện và dự báo lỗi đĩa cứng trong các mạng chuyên dụng như mạng quân sự của Binh chủng Thông tin liên lạc, Bộ Quốc phòng.

1.1. Bài toán phát hiện và dự báo lỗi đĩa cứng trong các mạng chuyên dụng

Trong các hệ thống mạng chuyên dụng (mạng quân sự, mạng điều khiển công nghiệp SCADA, ...), việc đảm bảo độ tin cậy, tính bảo mật cao và tính sẵn sàng của hệ thống lưu trữ là tối quan trọng. Ổ đĩa cứng (HDD hoặc SSD) là một thành phần chính trong hầu hết các hệ thống lưu trữ dữ liệu hiện đại. Tuy nhiên, ổ đĩa cứng cũng là một trong những thiết bị dễ hỏng nhất theo thời gian và khi phát sinh hỏng hóc sẽ gây thiệt hại lớn nhất, khó khôi phục hệ thống nhất và làm gián đoạn các hoạt động hệ thống lớn nhất so với hỏng hóc các thành phần khác. Vì vậy, việc phát hiện sớm và dự báo lỗi ổ đĩa cứng đóng vai trò quan trọng nhằm giảm thiểu thời gian chết của hệ thống, bảo vệ dữ liệu và hỗ trợ bảo trì chủ động.

1.2. Đặc điểm cấu trúc ổ cứng, cơ chế bảo vệ dữ liệu ổ cứng

1.2.1. Đặc điểm cấu trúc các ổ cứng phổ biến hiện nay

Trên thị trường hiện nay có nhiều loại ổ đĩa cứng của nhiều hãng khác nhau, một số hãng lớn như Seagate Technology (Mỹ), Western Digital – WD (Mỹ), Toshiba (Nhật Bản), Samsung (Hàn Quốc), Intel, Itachi, ... Đi cùng với đó là nhiều công nghệ sản xuất ổ đĩa cứng khác nhau. Một số công nghệ ổ đĩa cứng phổ biến hiện nay gồm:

Ổ cứng HDD (Hard Disk Drive): Thành phần chính gồm một hay nhiều đĩa từ (platter) được phủ vật liệu từ tính, gắn trên trục quay; đầu đọc/ghi gắn trên cần di chuyển (actuator arm). Khi ghi, từ tính của bề mặt đĩa được thay đổi để lưu thông tin. Khi đọc, đầu từ cảm nhận thay đổi từ tính. Thường được sử dụng phổ biến trong lưu trữ dung lượng lớn, như NAS, máy chủ backup, giám sát camera, ...

Ổ cứng SSD (Solid State Drive): Về cấu trúc không có bộ phận chuyển động; dữ liệu được lưu trữ trên chip flash NAND. Gồm các thành phần: controller (bộ điều khiển), DRAM cache và nhiều cell NAND chia theo SLC, MLC, TLC, QLC. Được sử dụng phổ biến trong máy chủ ứng dụng, database tốc độ cao, các hệ thống yêu cầu truy cập IOPS cao, máy tính cá nhân và máy trạm chuyên dụng.

Ổ cứng Enterprise: Các ổ cứng dành riêng cho máy chủ (enterprise-grade) được tối ưu với Độ bền cao (MTBF > 2 triệu giờ), Hoạt động 24/7, Có firmware chuyên biệt để tương

thích RAID, Giao tiếp SAS, SATA hoặc NVMe chuyên dụng

1.2.2. Các cơ chế bảo vệ dữ liệu trên ổ cứng

Các ổ đĩa cứng được tích hợp hỗ trợ nhiều giải pháp kỹ thuật để tăng độ bền và bảo vệ dữ liệu. Các cơ chế để bảo vệ dữ liệu phổ biến hiện nay gồm:

- **RAID (Redundant Array of Independent Disks):** Là kỹ thuật kết hợp nhiều ổ cứng vật lý thành một đơn vị logic để tăng hiệu suất, độ tin cậy hoặc cả hai. Các loại RAID 0, RAID 1, RAID 5, RAID 6, RAID 10.

- **Snapshot và Clone:** Lưu ảnh chụp nhanh trạng thái dữ liệu tại thời điểm nhất định (snapshot) hoặc tạo bản sao hoàn chỉnh ổ đĩa (clone), được sử dụng trong môi trường ảo hóa (VMware, Hyper-V) hoặc hệ điều hành hỗ trợ snapshot (Windows Volume Shadow Copy).

- **ECC và CRC:** Các ổ đĩa cứng hiện đại thường được tích hợp sẵn cơ chế tự sửa lỗi ở mức bit ECC (Error Correction Code) và cơ chế kiểm tra dữ liệu ghi/đọc CRC (Cyclic Redundancy Check)

1.3. Khái quát về các kỹ thuật, công nghệ sử dụng để theo dõi giám sát hoạt động của ổ cứng

1.3.1. Công nghệ SMART

SMART (viết tắt của *Self-Monitoring, Analysis and Reporting Technology*) là một công nghệ tích hợp bên trong ổ cứng HDD, SSD hoặc ổ đĩa lai SSHD nhằm tự động giám sát trạng thái hoạt động và dự đoán các lỗi tiềm ẩn có thể xảy ra trong tương lai. Đây là chuẩn công nghiệp được hỗ trợ bởi hầu hết các nhà sản xuất ổ đĩa hiện nay như Seagate, Western Digital, Toshiba, Samsung, v.v.

SMART hoạt động bằng cách liên tục ghi nhận và cập nhật các chỉ số kỹ thuật trong quá trình ổ đĩa vận hành. Dựa vào các thông số này, người dùng và hệ thống có thể đánh giá tình trạng "sức khỏe" của ổ đĩa, từ đó chủ động sao lưu dữ liệu hoặc thay thế ổ đĩa trước khi xảy ra sự cố nghiêm trọng.

Một số chỉ số chính được nêu Bảng 1.2

Bảng 1.2. Các thuộc tính chỉ số SMART chính

ID	Attributes	Chỉ số	Mô tả
1	Raw Read Error Rate	Tỷ lệ lỗi đọc thô	Đã xảy ra lỗi khi đọc dữ liệu thô từ đĩa
			Chỉ ra sự cố với bề mặt đĩa hoặc đầu đọc/ghi.
			<i>Thuộc tính quan trọng</i>
3	Spin Up Time	Thời gian quay	Thời gian cần thiết để trục chính quay đến vòng tua tối đa
			Chỉ ra sự cố với động cơ hoặc ổ trục.

ID	Attributes	Chỉ số	Mô tả
5	Reallocated Sector Count (Reallocated Sectors Count)	Số lượng khu vực được phân bổ lại (Số lượng khu vực được phân bổ lại)	<i>Thuộc tính quan trọng</i>
			Giá trị này không ảnh hưởng trực tiếp đến tình trạng của ổ đĩa.
			Số lượng sector được di chuyển đến vùng dự phòng
			Chỉ ra sự cố với bề mặt đĩa hoặc đầu đọc/ghi.
7	Seek Error Rate	Tìm kiếm tỷ lệ lỗi	<i>Thuộc tính quan trọng</i>
			Chức năng chính xác của thuộc tính này không được chỉ định.
			Tỷ lệ lỗi định vị của đầu đọc/ghi
			Chỉ ra vấn đề với servo, đầu. Nhiệt độ cao cũng có thể gây ra vấn đề này.
8	Seek Time Performance	Tìm kiếm hiệu suất thời gian	<i>Thuộc tính quan trọng</i>
			Thời gian trung bình của các hoạt động tìm kiếm của đầu
			Chỉ ra sự cố với servo.
			Đơn vị đo phụ thuộc vào nhà sản xuất.
191	Mechanical Shock	Sốc cơ học	Số lượng các vấn đề gây ra bởi sốc cơ học
			Tăng tốc (ví dụ như ngã) có thể gây ra sốc cơ học.
194	HDD Temperature	Nhiệt độ ổ cứng	Nhiệt độ đĩa
			Nhiệt độ bên trong vỏ ổ cứng.
			Số lỗi được sửa bằng cơ chế sửa lỗi nội bộ.
197	Current Pending Sector Count	Số lượng ngành đang chờ xử lý hiện tại	Số lượng sector không ổn định
			Các sector đang chờ xử lý này có thể được ánh xạ lại vào vùng dự phòng.
			<i>Thuộc tính quan trọng</i>
198	Off-Line Uncorrectable Sector Count	Số lượng Sector không thể sửa lỗi ngoại tuyến	Số lượng lỗi không thể sửa được khi đọc/ghi
			Chỉ ra sự cố với bề mặt đĩa hoặc đầu đọc/ghi.
			<i>Thuộc tính quan trọng</i>
199	Ultra ATA CRC Error Count	Đếm lỗi CRC Ultra ATA	Số lượng lỗi trong quá trình truyền dữ liệu giữa đĩa và máy chủ
			Chỉ ra sự cố với nguồn điện hoặc cáp dữ liệu.
221	G-Sense Error Rate	Tỷ lệ lỗi G-Sense	Số lỗi cơ học
			Số lỗi do va chạm hoặc rung động.
222	Loaded Hours	Giờ đã tải	Số giờ bật nguồn
			Giá trị này liên tục tăng (mỗi giờ một lần).
223	Load/Unload Retry Count	Tải/Gỡ bỏ số lần thử lại	Số lượng hoạt động tải/dỡ
			Số lượng đầu truyền động đi vào/rời khỏi vùng dữ liệu.
224	Load Friction	Tải Ma Sát	Tỷ lệ ma sát cơ học

ID	Attributes	Chỉ số	Mô tả
			Tỷ lệ ma sát giữa các bộ phận cơ khí. Chỉ ra vấn đề với hệ thống cơ khí của ổ đĩa.
226	Load-in Time	Thời gian tải vào	Tổng thời gian đầu đọc được tải
			Thời gian đầu đọc/ghi ở trong vùng dữ liệu.
231	Hard Disk Temperature	Nhiệt độ ổ cứng	Nhiệt độ đĩa
			Nhiệt độ bên trong vỏ ổ cứng.

(Nguồn: <https://www.hdsentinel.com/smart/smartattr.php>)

Mỗi chỉ số SMART có 2 giá trị chính:

- Normalized: giá trị đã chuẩn hóa (thường từ 1–100 hoặc 1–200)
- Raw: giá trị thô đo được (ví dụ: số sector lỗi thực tế)

1.3.2. Công cụ giám sát chuyên dụng

Hệ điều hành Windows: Một số ứng dụng được tích hợp, cài đặt trên hệ điều hành Windows để theo dõi tình trạng, thông số ổ cứng như ứng dụng CrystalDiskInfo, Hard Disk Sentinel dùng để theo dõi trạng thái SMART, nhiệt độ, tuổi thọ SSD,...

Hệ điều hành Linux: Một số ứng dụng như *smartmontools* (công cụ CLI mạnh mẽ để kiểm tra SMART và tự động gửi cảnh báo), *iostat*, *hdparm* (Phân tích hiệu năng I/O, thông tin phân cứng ổ đĩa)

1.3.3. Hệ thống giám sát tập trung

Các giải pháp như PRTG, Zabbix, Nagios, Netdata, ... hỗ trợ việc thu thập log từ ổ đĩa thông qua các bộ cảm biến (sensor), phân tích và đưa ra cảnh báo về nhiệt độ, một số bất thường trong đọc/ghi dữ liệu hay cảnh báo về mức độ sử dụng ổ đĩa cứng, ... Các thông tin được hiển thị dưới dạng các biểu đồ trực quan (dashboard), các báo cáo cận thời gian thực.

1.3.4. Công nghệ giám sát mức phần cứng – firmware tích hợp

Các ổ enterprise (Seagate Exos, WD Ultrastar, v.v.) thường được tích hợp sẵn firmware hỗ trợ theo dõi mức độ mài mòn (wear level), block hỏng, ECC error rate, v.v Một số dòng SSD NVMe hỗ trợ NVMe SMART log và telemetry log giúp theo dõi tình trạng ổ đĩa cứng qua các lệnh truy vấn (command line) hoặc qua các giao diện lập trình (API) được các hãng cung cấp.

1.4. Các công trình nghiên cứu liên quan

1.4.1. Các nghiên cứu theo hướng giám sát, thống kê các chỉ số SMART

Tính xác suất thống kê theo các chỉ số SMART:

Nhóm của Google nghiên cứu xu thế lỗi của một lượng lớn ổ cứng trong trung tâm dữ liệu. Các tác giả đã thực hiện thu thập và phân tích hành vi hỏng hóc của hơn 100.000 ổ đĩa cứng (HDD) trong trung tâm dữ liệu của Google. Phương pháp sử dụng là xác suất thống kê để phân tích mối tương quan giữa các chỉ số SMART và xác suất hỏng hóc. Đây là một trong những nghiên cứu đầu tiên với dữ liệu quy mô lớn và theo dõi chi tiết các chỉ số SMART, hoạt động và nhiệt độ của ổ đĩa. Mục tiêu của nghiên cứu này là xác định các yếu tố ảnh hưởng đến độ bền của ổ cứng. Các tác giả đã chỉ ra cách thức Google xây dựng hệ thống thu thập dữ liệu thời gian thực từ các máy chủ, sử dụng Bigtable và MapReduce để phân tích. Kết quả đạt được của nghiên cứu bao gồm:

- Tỷ lệ hỏng hóc trung bình: từ 1.7% năm đầu lên 8.6% vào năm thứ 3.
- Không tìm thấy mối liên hệ mạnh giữa nhiệt độ cao hoặc mức độ sử dụng cao với hỏng hóc.
- Một số thông số SMART có tương quan cao với lỗi, nhưng vẫn không đủ để dự đoán chính xác.

Các kết quả này được thể hiện trên hình 1.5.

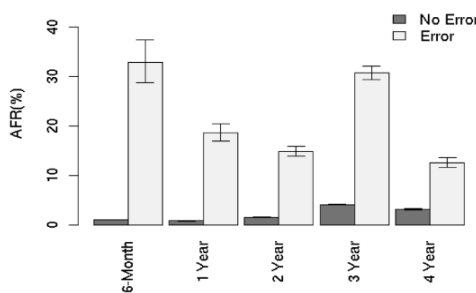


Figure 6: AFR for scan errors.

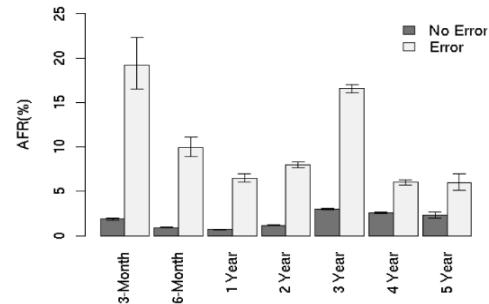
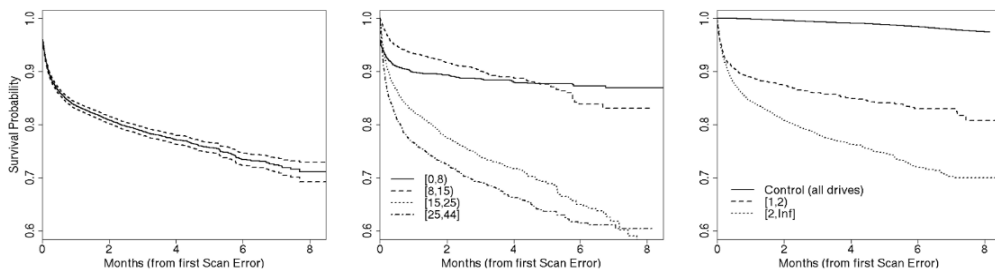


Figure 7: AFR for reallocation counts.



Hình 1.5. Một số kết quả nghiên cứu của Google

1.4.2. Các nghiên cứu theo hướng sử dụng học máy, học sâu

Nghiên cứu của nhóm tác giả Gargiulo et.al năm 2021 tập trung vào vấn đề dự đoán lỗi ổ cứng thông qua học máy với việc gán nhãn tự động. Nghiên cứu đã sử dụng hơn 65.000 ổ đĩa được thu thập tại trung tâm dữ liệu CERN. Mô hình học máy được sử dụng cho thử nghiệm là Regularized Greedy Forest - RGF (một biến thể của Gradient Boosted Decision

Trees). Mô hình có sử dụng thêm tham số tham chiếu (dạng regularization) để tăng độ ổn định, kết hợp chiến lược chọn tham số tối ưu bằng thực nghiệm (Design of Experiment - DOE).

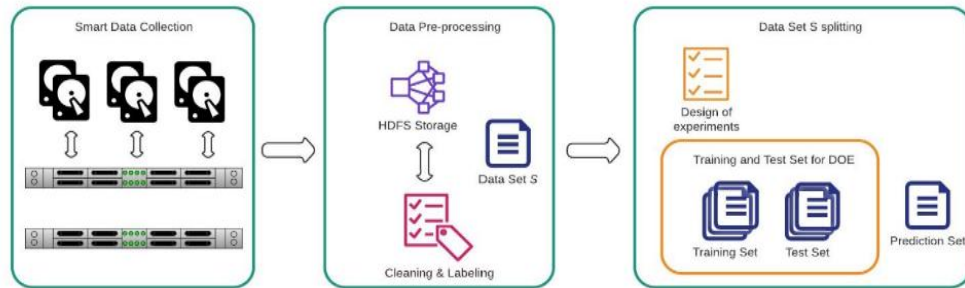
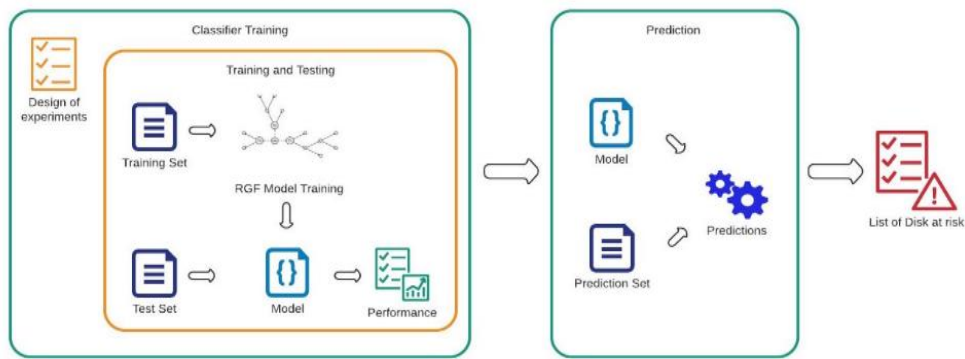


Figure 1. Data Collection and pre-processing phases.



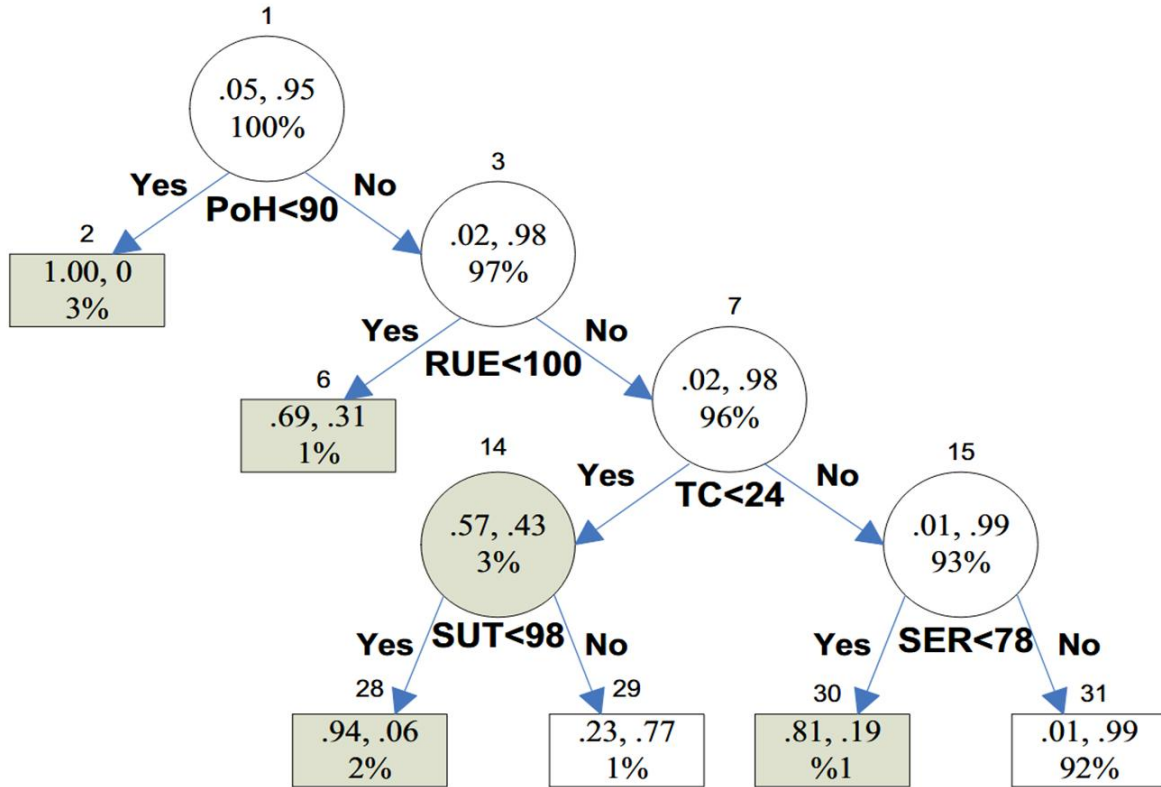
Hình 1.6. Mô hình dự đoán của nghiên cứu của nhóm CERN [4, tr.4]

Kết quả nghiên cứu được nêu ở Bảng 1.4 cho thấy các tỷ lệ Recall, FPR và LR (Recall / FPR) đạt được trong ba loại mô hình sử dụng trong nghiên cứu.

Bảng 1.4. Kết quả nghiên cứu của nhóm CERN

Model	Recall	FPR (False Positive Rate)	LR+ (Recall/FPR)
Ischia	98.4%	0.2%	659
Capri	92.2%	0.8%	115
Ponza	78.1%	0.9%	86

Nhóm nghiên cứu của Li Jing đề xuất mô hình dự đoán lỗi ổ cứng sử dụng cây phân loại và hồi quy (Classification & Regression Trees – CART). Dữ liệu thực tế từ trung tâm dữ liệu với 25.792 ổ đĩa, sử dụng 13 đặc trưng SMART gồm: chỉ số lỗi đọc, thời gian bật máy, lỗi không thể sửa, nhiệt độ, tốc độ thay đổi, v.v. Nhóm nghiên cứu đã áp dụng cây phân loại (CT) để phân biệt ổ tốt / lỗi và cây hồi quy (RT) để đánh giá "mức độ sức khỏe" của ổ cứng.



Hình 1.7. Mô hình phân loại của nghiên cứu CART

Kết quả nghiên cứu cho thấy, với mô hình phân loại (CT) có thể dự đoán được >95% lỗi ổ cứng, tỷ lệ cảnh báo sai (FAR) đạt được <0.1%. Mô hình có thể phát hiện lỗi trung bình trước khi xảy ra sự cố 2 tuần. Với mô hình hồi quy (RT), hệ thống cho phép xử lý ưu tiên theo mức độ nguy cơ và điều chỉnh linh hoạt giữa FDR – FAR bằng ngưỡng và đánh giá được mức độ rủi ro của ổ cứng (gần hỏng hay còn ổn).

Nghiên cứu của Wang và cộng sự năm 2023 tập trung vào vấn đề tối ưu hiệu quả học máy cho dự đoán lỗi ổ cứng bằng cách sử dụng lựa chọn các đặc trưng dựa vào phân lớp. Mục tiêu của nghiên cứu là cải thiện tốc độ và độ chính xác trong dự đoán lỗi ổ cứng (HDD) sử dụng mô hình học máy. Bài báo tập trung giải quyết vấn đề về thời gian huấn luyện lâu và độ trễ dự đoán cao trong bối cảnh dữ liệu SMART cập nhật liên tục. Các tác giả đề xuất giải pháp chọn đặc trưng 2 lớp (two-layer feature selection) nhằm giảm số lượng đặc trưng đầu vào mà không làm giảm hiệu suất dự đoán.

Nghiên cứu sử dụng dữ liệu từ Backblaze (2020) gồm 2 loại ổ cứng ST4000DM000 và ST12000NM0007, thử nghiệm với nhiều mô hình khác nhau (Naïve Bayes, Random Forest, SVM, GBDT, CNN và LSTM). Kết quả nghiên cứu được nêu ở Bảng 1.5

Bảng 1.5. Kết quả nghiên cứu của Wang

		DM000									NM0007								
		Group1			Group2_Prs			Group2_Sprm			Group1			Group2_Prs			Group2_Sprm		
		Pre	Re	F1	Pre	Re	F1	Pre	Re	F1	Pre	Re	F1	Pre	Re	F1	Pre	Re	F1
Original	NB	0.69	0.38	0.49	0.7	0.37	0.49	0.72	0.37	0.48	0.79	0.31	0.45	0.81	0.27	0.41	0.8	0.3	0.43
	RF	0.99	0.98	0.99	0.99	0.99	0.99	1	0.98	0.99	1	0.97	0.98	0.99	0.94	0.97	1	0.95	0.97
	SVM	0.91	0.54	0.68	0.89	0.09	0.17	0.94	0.46	0.62	0.84	0.31	0.45	0.83	0.29	0.43	0.77	0.24	0.36
	GBDT	0.97	0.79	0.87	0.95	0.76	0.85	0.97	0.77	0.86	0.93	0.71	0.81	0.92	0.61	0.73	0.91	0.68	0.78
	CNN	0.9	0.62	0.74	0.8	0.54	0.65	0.77	0.54	0.64	0.82	0.35	0.49	0.86	0.36	0.51	0.74	0.33	0.46
	LSTM	0.98	0.9	0.94	0.96	0.95	0.96	0.99	0.91	0.95	0.91	0.92	0.92	0.94	0.92	0.93	0.96	0.81	0.91
Pearson	NB	0.78	0.36	0.49	0.78	0.35	0.48	0.78	0.35	0.48	0.78	0.3	0.43	0.81	0.26	0.4	0.79	0.29	0.42
	RF	0.95	0.46	0.62	0.95	0.46	0.62	0.95	0.47	0.63	0.95	0.72	0.82	0.89	0.57	0.69	0.94	0.72	0.82
	SVM	0.78	0.24	0.36	0.75	0.12	0.21	0.77	0.12	0.21	0.77	0.32	0.46	0.82	0.31	0.45	0.73	0.25	0.37
	GBDT	0.92	0.43	0.59	0.9	0.42	0.57	0.92	0.43	0.59	0.9	0.62	0.74	0.87	0.54	0.67	0.9	0.62	0.74
	CNN	0.72	0.34	0.47	0.77	0.24	0.36	0.8	0.2	0.32	0.72	0.4	0.51	0.77	0.36	0.49	0.68	0.37	0.48
	LSTM	0.94	0.36	0.52	0.87	0.37	0.52	0.87	0.37	0.52	0.95	0.48	0.64	0.88	0.46	0.6	0.88	0.46	0.6
Spearman	NB	0.73	0.35	0.47	0.74	0.34	0.47	0.74	0.34	0.47	0.78	0.3	0.43	0.81	0.26	0.4	0.79	0.29	0.42
	RF	0.99	0.49	0.66	0.98	0.48	0.64	0.96	0.5	0.65	0.96	0.71	0.81	0.91	0.56	0.7	0.95	0.72	0.82
	SVM	0.78	0.23	0.36	0.74	0.12	0.2	0.76	0.11	0.2	0.77	0.32	0.46	0.82	0.31	0.45	0.73	0.25	0.37
	GBDT	0.95	0.42	0.59	0.93	0.41	0.57	0.95	0.42	0.59	0.89	0.62	0.73	0.87	0.54	0.67	0.9	0.62	0.74
	CNN	0.71	0.35	0.47	0.77	0.29	0.42	0.77	0.24	0.36	0.77	0.33	0.46	0.73	0.38	0.5	0.7	0.33	0.45
	LSTM	0.95	0.23	0.37	0.68	0.29	0.41	0.78	0.29	0.42	0.87	0.51	0.65	0.91	0.45	0.61	0.85	0.54	0.66
J-Index	NB	0.62	0.36	0.46	0.63	0.35	0.45	0.71	0.3	0.43	0.77	0.32	0.45	0.77	0.32	0.46	0.75	0.3	0.43
	RF	0.98	0.93	0.96	0.98	0.91	0.95	0.97	0.89	0.93	0.98	0.88	0.93	0.97	0.84	0.9	0.98	0.84	0.91
	SVM	0.81	0.3	0.44	0.77	0.15	0.25	0.79	0.15	0.25	0.88	0.34	0.49	0.88	0.35	0.5	0.8	0.22	0.35
	GBDT	0.88	0.68	0.76	0.87	0.68	0.76	0.87	0.68	0.76	0.89	0.69	0.78	0.87	0.69	0.77	0.9	0.68	0.78
	CNN	0.79	0.47	0.59	0.54	0.65	0.59	0.57	0.62	0.59	0.66	0.54	0.6	0.71	0.44	0.54	0.54	0.5	0.52
	LSTM	0.94	0.83	0.88	0.94	0.81	0.87	0.88	0.75	0.81	0.83	0.73	0.78	0.88	0.73	0.8	0.85	0.67	0.75
Entropy	NB	0.63	0.37	0.47	0.7	0.37	0.49	0.71	0.36	0.47	0.72	0.43	0.54	0.78	0.31	0.45	0.75	0.32	0.44
	RF	1	0.98	0.99	0.99	0.98	0.99	0.99	0.99	0.99	1	0.97	0.98	1	0.97	0.98	1	0.97	0.98
	SVM	0.94	0.54	0.69	0.82	0.21	0.33	0.93	0.54	0.68	0.91	0.42	0.57	0.85	0.33	0.47	0.8	0.26	0.4
	GBDT	0.96	0.78	0.86	0.95	0.75	0.84	0.96	0.77	0.85	0.93	0.72	0.81	0.91	0.6	0.72	0.92	0.65	0.76
	CNN	0.78	0.76	0.77	0.86	0.7	0.77	0.86	0.69	0.76	0.85	0.39	0.54	0.36	0.56	0.44	0.63	0.44	0.52
	LSTM	0.99	0.98	0.99	0.99	0.96	0.97	0.96	0.85	0.9	0.9	0.92	0.91	0.95	0.88	0.91	0.99	0.94	0.96
Gini	NB	0.7	0.37	0.49	0.76	0.35	0.48	0.77	0.35	0.48	0.75	0.4	0.52	0.78	0.31	0.44	0.75	0.32	0.44
	RF	0.99	0.99	0.99	1	0.99	0.99	0.99	0.97	0.98	1	0.97	0.98	1	0.95	0.97	1	0.96	0.98
	SVM	0.94	0.51	0.66	0.92	0.52	0.67	0.92	0.51	0.66	0.89	0.41	0.56	0.84	0.31	0.45	0.78	0.24	0.37
	GBDT	0.96	0.78	0.86	0.96	0.77	0.85	0.97	0.96	0.85	0.91	0.71	0.8	0.92	0.6	0.73	0.92	0.68	0.78
	CNN	0.81	0.71	0.76	0.87	0.73	0.79	0.83	0.78	0.8	0.36	0.72	0.48	0.77	0.42	0.54	0.76	0.36	0.49
	LSTM	1	0.97	0.98	0.99	0.98	0.98	1	0.96	0.98	1	0.98	0.99	1	0.99	0.99	0.98	0.92	0.95

1.4.3. Nguồn dữ liệu SMART của Backblaze

Backblaze là một công ty cung cấp dịch vụ sao lưu đám mây (cloud backup), sở hữu hàng trăm ngàn ổ đĩa trong trung tâm dữ liệu. Kể từ năm 2014, họ bắt đầu công khai dữ liệu SMART (Hơn 70 chỉ số ổ cứng bao gồm smart_x_normalized, smart_x_raw, model, serial_number, failure, v.v.) theo thời gian thực với trên 250.000 loại ổ đĩa cứng từ nhiều hãng như Seagate, Western Digital, Toshiba, HGST, v.v.

Backblaze cũng triển khai nhiều nghiên cứu nhằm phát hiện sớm lỗi ổ cứng. Một số kết quả nghiên cứu của Backblaze được nêu ở Bảng 1.3.

Bảng 1.3. Dữ liệu thu thập của Backblaze

Năm	Số ổ đĩa theo dõi	Tổng dung lượng	Tỉ lệ hỏng trung bình (AFR)
2019	~120,000	>600 PB	~1.89%
2022	>200,000	>1.2 EB	~1.37%
2024	~270,000+	>1.6 EB	~1.22%

Các kết quả nghiên cứu của Backblaze đã chỉ ra rằng: (1) Không phải tất cả lỗi đều có cảnh báo SMART rõ ràng: Có những ổ đĩa lỗi đột ngột mà không có sự thay đổi rõ rệt về SMART. Điều này thách thức các mô hình truyền thống chỉ dựa trên ngưỡng SMART cố định. (2) SMART_5 (Reallocated Sectors) là chỉ số SMART dẫn đến lỗi ổ đĩa cứng cao nhất. (3) Số giờ hoạt động không tỉ lệ thuận với lỗi.

1.5. Khái quát về mạng máy tính quân sự của Binh chủng Thông tin liên lạc, Bộ Quốc phòng và nhu cầu phát hiện, dự báo lỗi ổ cứng

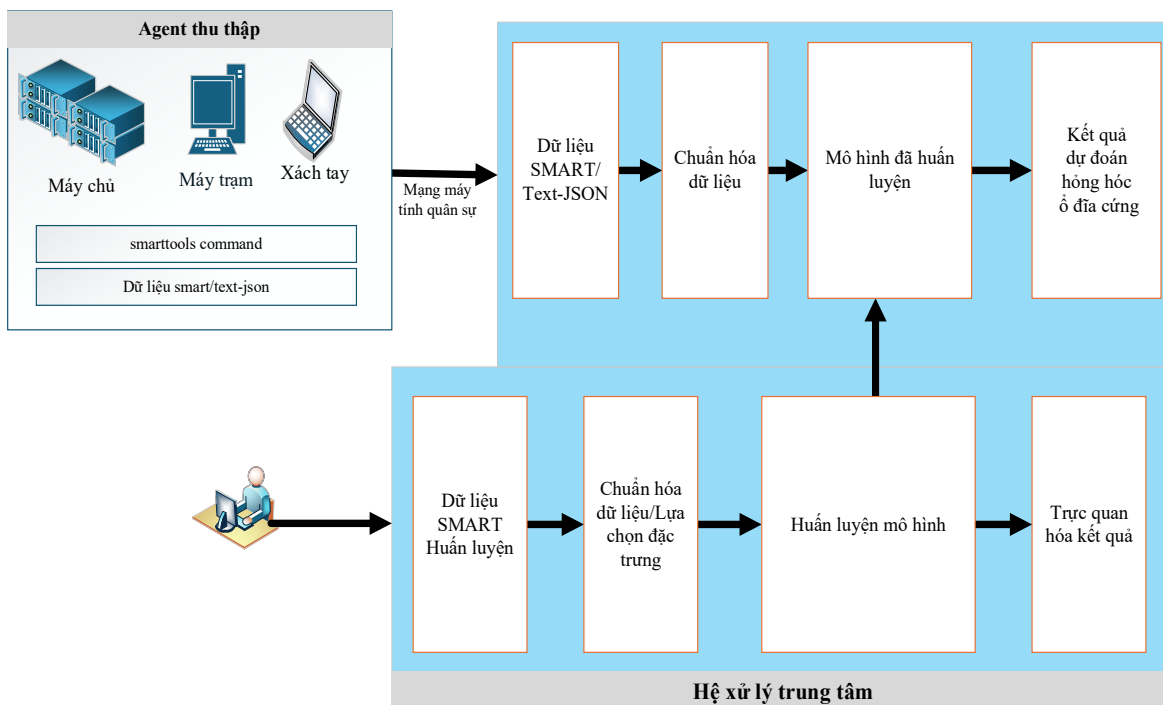
Mạng máy tính quân sự tại Binh chủng Thông tin liên lạc được tổ chức gồm 10 hệ thống mạng LAN tại các cơ quan, đơn vị kết nối qua hệ thống mạng truyền số liệu. Trong hệ thống mạng có hơn 2600 máy tính kết nối, 01 trung tâm dữ liệu, 10 phòng máy chủ với gần 100 máy chủ vật lý. Do đặc điểm về yêu cầu bảo mật thông tin nên hệ thống mạng hoàn toàn độc lập, không kết nối liên thông với bất kỳ hệ thống mạng nào khác (mạng internet, mạng doanh nghiệp, ...). Cho tới nay, công tác theo dõi, phát hiện lỗi ổ cứng trên mạng vẫn chủ yếu theo hình thức thủ công bằng các công cụ tiện ích của hệ điều hành, chưa có một hệ thống phát hiện lỗi ổ cứng được ứng dụng trong mạng máy tính quân sự của Binh chủng Thông tin liên lạc. Điều này gây khó khăn cho việc sớm phát hiện lỗi, dự đoán lỗi và lập kế hoạch bảo trì thiết bị. Những vấn đề kỹ thuật đặt ra là: các lỗi ổ cứng rất đa dạng, do nhiều nguyên nhân nên rất khó phát hiện kịp thời, chưa có một giải pháp hiệu quả để phát hiện và dự đoán lỗi, việc sử dụng các phần mềm giám sát thiết bị hiện có còn hạn chế về tính năng dự báo và phát hiện sớm, giải pháp đưa ra cần xem xét đặc thù của mạng máy tính chuyên dụng có tính đóng của quân sự.

Chương 2: NGHIÊN CỨU XÂY DỰNG HỆ THỐNG PHÁT HIỆN, DỰ BÁO LỖI Ổ CỨNG VỚI MÔ HÌNH HỌC MÁY

Nội dung chương 2 trình bày về đề xuất mô hình, kiến trúc hệ thống phát hiện, dự báo lỗi ổ cứng trong mạng máy tính quân sự chuyên dụng của Binh chủng Thông tin liên lạc, Bộ Quốc phòng. Các nội dung chính gồm: xây dựng mô hình hệ thống; các thành phần của hệ xử lý trung tâm, tập trung vào các vấn đề về thu thập, phân tích và chuẩn hóa dữ liệu SMART; thực hiện cân bằng dữ liệu; lựa chọn mô hình học máy phù hợp; cách thức huấn luyện và dự báo; phương pháp đánh giá mô hình.

2.1. Xây dựng mô hình hệ thống

Kiến trúc một hệ thống giám sát thiết bị nói chung thường gồm hai hệ chính: hệ Agent thu thập dữ liệu từ thiết bị và hệ xử lý trung tâm thực hiện xử lý, phân tích, đưa ra kết quả cảnh báo. Trên cơ sở đó, mô hình hệ thống phát hiện lỗi ổ cứng trong mạng quân sự được học viên đề xuất như trên Hình 2.1 với 02 thành phần chính: Agent thu thập và hệ xử lý trung tâm.



Hình 2.1. Mô hình tổng quan hệ thống phát hiện hỏng hóc trong mạng quân sự

- Hệ Agent thu thập dữ liệu SMART của ổ cứng: Được triển khai trên máy chủ, máy trạm, máy tính người dùng trong mạng để thu thập các giá trị SMART của ổ đĩa cứng, đóng gói và truyền dữ liệu về máy chủ xử lý.

- Hệ xử lý trung tâm: Thực hiện chuẩn hóa dữ liệu do các Agent đẩy lên, phân tích, dự đoán tình trạng hỏng hóc của ổ đĩa cứng bằng các mô hình đã được huấn luyện, đưa ra kết

quả dự đoán (phần trăm tốt/hỏng) của ổ đĩa cứng). Đối với các loại ổ cứng chưa được huấn luyện, người quản trị có thể thu thập dữ liệu (thủ công) và huấn luyện mô hình.

2.2. Hệ Agent thu thập dữ liệu SMART của ổ cứng

Nhiệm vụ của hệ Agent: Thu thập dữ liệu SMART của ổ đĩa cứng trên máy chủ, máy trạm, máy tính người dùng trong mạng máy tính quân sự. Thực hiện đóng gói và đẩy dữ liệu về máy chủ xử lý phân tích qua mạng máy tính quân sự.

Hệ điều hành Windows: Sử dụng công cụ smartmontools để thu thập. Chương trình thực thi lấy dữ liệu SMART về đẩy dữ liệu về máy chủ xử lý:

```
@echo off
# Lấy địa chỉ IP của máy
for /f "tokens=2 delims=: " %%A in ('ipconfig ^| findstr /i "IPv4") do (
    set ip=%%A)
# Loại bỏ khoảng trắng
set ip=%ip: =%
set ip=%ip: .=%
# Lấy ngày giờ hiện tại
for /f "tokens=1-4 delims=/ " %%a in ("%date%") do (
    set dd=%%a
    set mm=%%b
    set yyyy=%%c )
for /f "tokens=1-2 delims=: " %%a in ("%time%") do (
    set hh=%%a
    set min=%%b )
# Đảm bảo 2 chữ số
if 1%hh% LSS 10 set hh=0%hh%
if 1%min% LSS 10 set min=0%min%
# Tạo tên file
set filename=%ip%_%%yyyy%%mm%%dd%_%%hh%%min%_smart.txt
# Ghi thông tin SMART
```

Kết quả chương trình đẩy dữ liệu SMART đang thu thập về máy chủ xử lý \\SERVER_NAME được lưu trữ tại thư mục Share_SMART trên máy chủ. File dữ liệu có định dạng *IP_Date_Time_smart.txt*. Sử dụng các công cụ có sẵn trên Windows để thực thi chương trình tự động theo chu kỳ được thiết lập trước (01 giờ, 01 ngày, ...)

Hệ điều hành Linux: Sử dụng ứng dụng smartmontools trên Linux để thực hiện thu thập dữ liệu SMART. Chương trình thực thi như sau:

```
# Lấy địa chỉ IP nội bộ
IP=$(hostname -I | awk '{print $1}' | tr ' ' '_')
# Lấy ngày giờ hiện tại
DATE=$(date +%Y%m%d)
TIME=$(date +%H%M)
# Tên file
FILENAME="$IP_${DATE}_${TIME}_smart.txt"
# Đường dẫn thư mục lưu (ví dụ thư mục chia sẻ NFS hoặc thư mục cục bộ)
OUTPUT_DIR="/mnt/shared_smart_data"
mkdir -p "$OUTPUT_DIR" # Tạo nếu chưa có
# Lấy dữ liệu SMART (ở /dev/sda) và ghi vào file
smartctl -a /dev/sda > "$OUTPUT_DIR/$FILENAME"
```

Kết quả chương trình đẩy dữ liệu SMART đang thu thập ra thư mục /mnt/shared_smart_data được đồng bộ với thư mục \\SERVER_NAME\ Share_SMART trên máy chủ xử lý. Chương trình thực thi tự động theo chu kỳ được thiết lập trước (01 giờ, 01 ngày, ...).

2.3. Hệ xử lý trung tâm

Tập trung các chức năng chính của hệ thống, bao gồm các quá trình chuẩn hóa dữ liệu SMART do các Agent đẩy lên, phân tích dữ liệu, đưa ra kết quả về khả năng hỏng hóc của ổ đĩa cứng. Ngoài ra đối với các model (loại) ổ cứng chưa được huấn luyện, quản trị hệ thống có thể tiến hành huấn luyện bổ sung mô hình.

2.3.1. Tiến trình thu thập và tiền xử lý dữ liệu SMART

2.3.1.1. Thu thập dữ liệu để huấn luyện

Để huấn luyện mô hình cho hệ thống, học viên đã sử dụng bộ dữ liệu của Backblaze thu thập được năm 2020 trở lại đây. Các dữ liệu này đã được gán nhãn faild (1) tương đương ổ đĩa hỏng và no_faild (0) ổ đĩa đang hoạt động. Dữ liệu được thu thập lưu trữ dưới dạng các file .csv.

2.3.1.2. Chuẩn hóa dữ liệu SMART

Các dữ liệu SMART của Backblaze thu thập có đặc điểm rất mất cân bằng, tỉ lệ giữa ổ cứng được gán nhãn hỏng - 1 và ổ đĩa không hỏng được gán nhãn - 0 có độ chênh lệch rất lớn (Dữ liệu thu thập năm 2021, cụ thể: Số lượng ổ cứng không hỏng = 66.665.387 mẫu, trong khi đó số lượng ổ cứng hỏng là 2.123 bản ghi chỉ bằng 0,003% mẫu hỏng).

Ngoài ra, trong dữ liệu SMART có nhiều trường dữ liệu không hữu ích đối với bài toán phân loại, cần phải được chuẩn hóa, loại bỏ trước khi đưa vào mô hình huấn luyện. Trong nghiên cứu, học viên đã chuẩn hóa các trường dữ liệu sau:

- Loại bỏ các trường mang ý nghĩa cung cấp thông tin như: date (ngày tháng thu thập dữ liệu SMART của ổ đĩa cứng), model (loại ổ cứng), serial_Number, capacity (Dung lượng ổ cứng).

Bảng 2.1. Một số trường dữ liệu SMART thu thập của Backblaze

date	serial_number	model	capacity_byt/failure	smart_1_normalized	smart_1_raw	smart_2_normalized	smart_2_raw	smart_3	smart_3	smart_4	smart_4	smart_5	smart_5	smart_7	smart_7	smart_8	smart_8	smart_9	
1/2/2021	ZLWEGG	ST12000NM001G	1.20001E+13	0	100	2104304		99	0	100	1	100	0	89722387109				95	
1/2/2021	Z3V3B2QN	ST4000DM000	4.00079E+12	0	118	173899416		91	0	100	19	100	0	87576938600				50	
1/2/2021	ZLW0GNE	ST12000NM001G	1.20001E+13	0	78	65992144		98	0	100	2	100	0	89777099064				94	
1/2/2021	ZLVXJQ3	ST12000NM0007	1.20001E+13	0	82	170070192		98	0	100	2	100	0	87475386677				80	
1/2/2021	ZLW18MKT	ST14000NM001G	1.40005E+13	0	80	103967288		95	0	100	4	100	0	81122738572				99	
1/2/2021	Z0H0A0CXF97G	TOSHIBA MG07ACA147A	1.40005E+13	0	100	0	100	0	100	7790	100	15	100	0	100	0	100	0	89
1/2/2021	ZAJFLEP	ST8000NM0055	8.00156E+12	0	75	31093384		96	0	100	3	100	0	87532198811				96	
1/2/2021	ZAJ6NQJR	ST8000NM0055	8.00156E+12	0	80	96039784		89	0	100	8	100	0	86338782263				64	
1/2/2021	ZLV02XWG	ST12000NM0007	1.20001E+13	0	79	71266064		89	0	100	10	100	0	87465232326				74	
1/2/2021	ZAL0V0V6	ST8000DM002	8.00156E+12	0	69	8090112		88	0	100	6	100	0	963689E+09				54	

- Dữ liệu SMART của Backblaze mỗi trường giá trị được lưu dưới dạng 02 giá trị Raw (giá trị thô đo được) và Normalized (giá trị đã được chuẩn hóa). Trong nghiên cứu của đề án chỉ xử lý trên các trường giá trị Raw, loại bỏ các cột dữ liệu Normalized.

- Loại bỏ các mẫu dữ liệu không được gán nhãn là hỏng hay tốt (trường failure = NaN), đồng thời loại bỏ các model ổ đĩa cứng chưa hỏng lần nào (tổng failure = 0).

- Đối với model ổ đĩa cứng sau khi thực hiện chuẩn hóa dữ liệu từ các bước ở trên, được lưu vào một file .csv để tiếp tục quá trình chuẩn hóa:

- + Loại bỏ tất cả các cột toàn giá trị NaN.
- + Loại bỏ các cột có độ lệch chuẩn = 0 (giá trị cột không thay đổi)
- + Loại bỏ các mẫu (dòng) chứa giá trị NaN (Giá trị không xác định)

2.3.1.3. Cân bằng dữ liệu SMART

Đối với dữ liệu SMART có tỉ lệ mất cân bằng dữ liệu lớn sẽ ảnh hưởng đến kết quả huấn luyện của mô hình. Để tăng độ chính xác, có thể sử dụng một số thuật toán cân bằng dữ liệu, trong nghiên cứu học viên sử dụng SMOTE (**Synthetic Minority Over-sampling Technique**) là một kỹ thuật sinh mẫu dữ liệu **giả lập** thuộc lớp thiểu số bằng cách **nội suy tuyến tính** giữa một điểm thiểu số và các điểm lân cận gần nhất trong không gian đặc trưng. Nguyên lý của SMOTE như sau:

- Chọn một mẫu thuộc lớp tối thiểu x
- Tìm k láng giềng gần nhất của x (thường $k=5$)
- Sinh thêm điểm mới bằng công thức: $x_n = x + \partial * (x_m - x)$. Trong đó x_n là một hàng xóm ngẫu nhiên và ∂ thuộc $[0, 1]$ là hệ số nội suy ngẫu nhiên.

2.3.2. Mô hình, thuật toán sử dụng cho hệ thống

Trong nghiên cứu này, học viên sử dụng thuật toán Random Forest (RF) và XGBClassifier (XGB) để xây dựng mô hình dự báo, phát hiện hỏng hóc ổ đĩa cứng.

Random Forest: Random Forest là thuật toán kết hợp (ensemble learning), xây dựng từ việc huấn luyện rất nhiều cây quyết định (Decision Tree), sau đó kết hợp kết quả dự đoán của các cây để đưa ra kết quả chính xác hơn và ổn định hơn. Random Forest sử dụng bagging (Bootstrap Aggregating) để huấn luyện song song các cây quyết định. Đối với bài toán phân loại, Random Forest sử dụng kết quả theo hình thức bỏ phiếu đa số (majority voting) từ các cây. Thuật toán Random Forest hoạt động theo các bước:

Bước 1: Bootstrapping dữ liệu: Random Forest chọn ngẫu nhiên một lượng lớn mẫu dữ liệu (sample) từ tập dữ liệu gốc, kích thước các mẫu bằng kích thước dữ liệu ban đầu, nhưng có lấy mẫu lặp lại (một số dữ liệu có thể xuất hiện nhiều lần trong một mẫu).

Bước 2: Xây dựng cây quyết định (Decision Tree): Với mỗi tập dữ liệu con đã tạo, Random Forest xây dựng một cây quyết định (Decision Tree). Điểm đặc biệt của Random Forest: mỗi lần phân chia (split) một nút trong cây quyết định, chỉ có một tập con các đặc trưng (features) được chọn ngẫu nhiên để tìm cách phân chia tốt nhất. Điều này gọi là random feature selection.

Bước 3: Kết hợp kết quả dự đoán cuối cùng được tổng hợp từ các cây quyết định con:

- Phân loại: kết quả là lớp (class) được đa số các cây dự đoán.
- Hồi quy: kết quả là trung bình giá trị được dự đoán từ các cây.

Một khó khăn khi áp dụng thuật toán Random Forest là phải tạo lập cấu hình cho các tham số. Thay vì phải thử thủ công từng giá trị tham số cấu hình, thuật toán **RandomizedSearchCV** có thể giúp tự động tìm ra cấu hình Random Forest tốt nhất. Điều này rất quan trọng nhằm tối ưu hiệu suất mô hình.

RandomizedSearchCV: Kỹ thuật tìm kiếm siêu tham số tốt nhất cho mô hình. RandomizedSearchCV sẽ chọn ngẫu nhiên n tổ hợp tham số từ không gian của mô hình. Dùng k -fold cross-validation để đánh giá từng tổ hợp. Trả về mô hình với tổ hợp tham số tốt nhất theo tiêu chí đánh giá (F1-score).

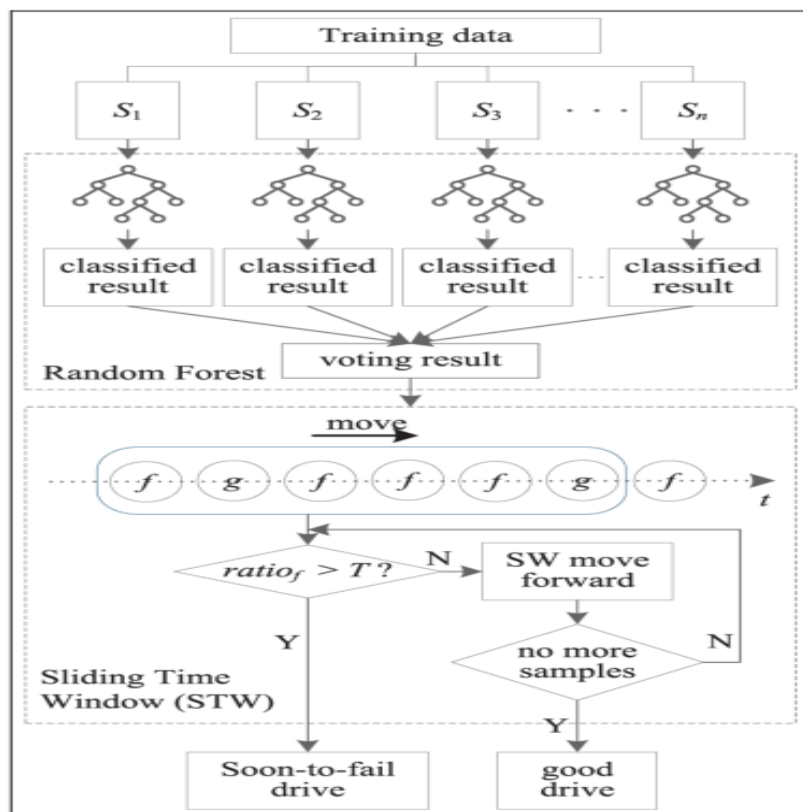
Mặt khác, một kỹ thuật có tên gọi là **Sliding Time Window (STW)** rất hay được kết hợp với Random Forest cho các tập dữ liệu thời gian (Time-series data), điển hình là các dữ liệu lỗi như lỗi ổ cứng. Random Forest không có khả năng xử lý dữ liệu chuỗi thời gian mà chỉ xử lý dữ liệu dạng bảng với các đặc trưng độc lập. STW sẽ giúp trích xuất các đặc trưng theo cửa sổ thời gian, biến dữ liệu chuỗi thời gian thành tập dữ liệu độc lập để Random Forest có thể xử lý. Ngoài ra, **cơ chế bỏ phiếu chọn lọc (Part-Voting)** có thể giúp nâng cao độ ổn định và tính chính xác trong phát hiện lỗi của hệ thống. **Random Forest kết hợp Sliding Time**

Window (STW) và cơ chế bỏ phiếu chọn lọc (Part-voting): Phương pháp dự đoán xem một ổ cứng có “sắp lỗi” không, bằng cách:

- Dùng nhiều cây quyết định (Random Forest) để phân loại từng mẫu dữ liệu SMART.
- Theo dõi chuỗi kết quả qua một cửa sổ thời gian trượt (Sliding Time Window).
- Nếu tỷ lệ cảnh báo lỗi vượt ngưỡng $\rightarrow$ kết luận “sắp lỗi”.

Thuật toán được mô tả như trong Hình 2.3 Lưu đồ phương pháp Random Forest kết hợp STW và cơ chế Part-voting. Trong đó:

- SS: chuỗi mẫu SMART của ổ HDD h (đầu vào)
- Trees: tập các cây trong Random Forest
- ws: kích thước cửa sổ thời gian
- T: ngưỡng tỷ lệ “fail” để kết luận ổ cứng sắp hỏng



Hình 2.3. Lưu đồ phương pháp Random Forest kết hợp STW và cơ chế Part-voting

XGBClassifier: Sử dụng thuật toán Gradient Boosting tối ưu cao để thực hiện bài toán phân loại. XGBClassifier xử lý tốt với bào toán mất cân bằng dữ liệu và kiểm soát overfitting tốt hơn Random Forest.

Gradient Boosting là một thuật toán học máy mạnh mẽ thuộc nhóm học có giám sát, cụ thể là kỹ thuật boosting. Gradient Boosting xây dựng mô hình mạnh bằng cách kết hợp

nhiều mô hình yếu (thường là cây quyết định nhỏ) một cách tuần tự. Gradient Boosting hoạt động qua các bước:

- Khởi tạo mô hình ban đầu: Mô hình đầu tiên dự đoán giá trị trung bình (hồi quy) hoặc log-odds (phân loại).
- Lặp qua nhiều vòng (rounds):
 - + Tính residuals (sai số còn lại): $\text{residual} = y_{\text{true}} - y_{\text{pred}}$
 - + Huấn luyện một mô hình mới để dự đoán residuals.
 - + Cập nhật mô hình tổng thể bằng cách cộng thêm mô hình mới nhân với learning rate.
- Tổng hợp tất cả các mô hình nhỏ để tạo mô hình cuối cùng.

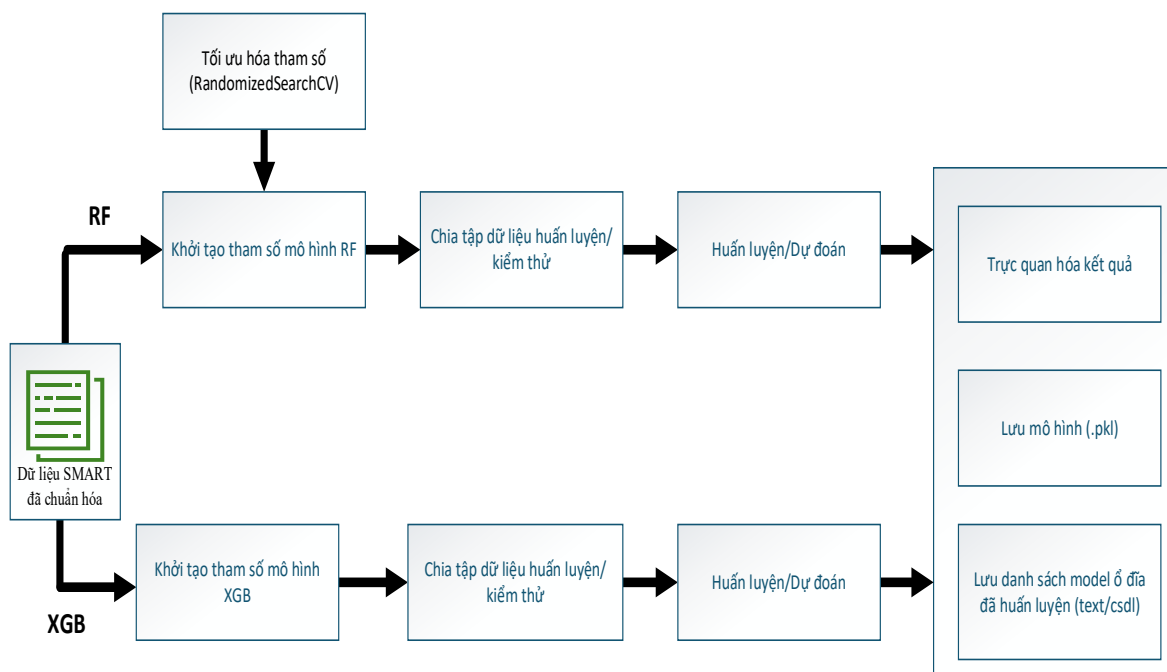
Các hàm tính toán trong Gradient Boosting: Hàm mất mát: $L(y, F(x))$ và Mô hình tổng: $F_m(x) = F_{m-1}(x) + \gamma_m h_m(x)$. Trong đó:

- $h_m(x)$ là cây quyết định nhỏ thứ m học theo gradient của hàm mất mát.
- γ_m là hệ số learning rate.
- $F_0(x)$ là mô hình khởi đầu.

Ta tìm $h_m(x)$ sao cho: $h_m(x) \approx -\nabla_{F_{m-1}} L(y, F_{m-1}(x))$

2.3.3. Tiến trình huấn luyện mô hình

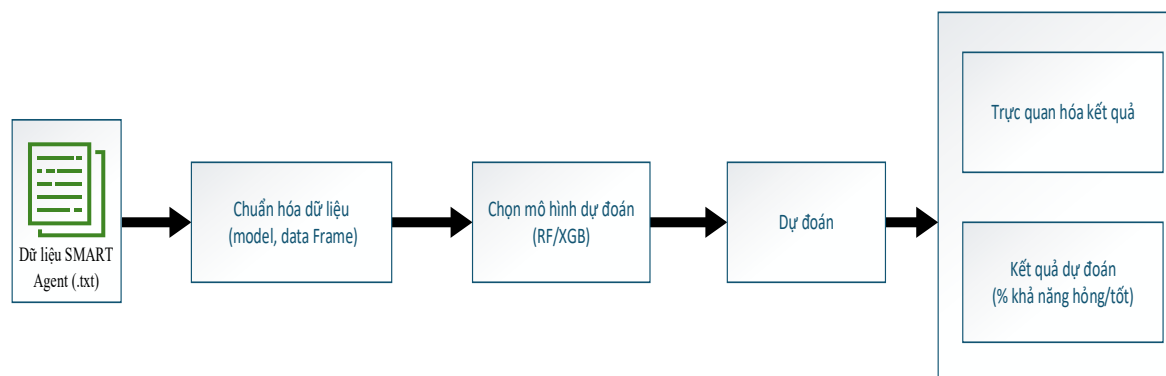
Tiến trình thực hiện được thể hiện ở Hình 2.5. Tiến trình huấn luyện mô hình RF và XGB gồm các bước: Đọc dữ liệu SMART, lựa chọn mô hình (RF hoặc XGB), Chia tập dữ liệu huấn luyện, Huấn luyện/Dự đoán, Lưu mô hình, trực quan kết quả.



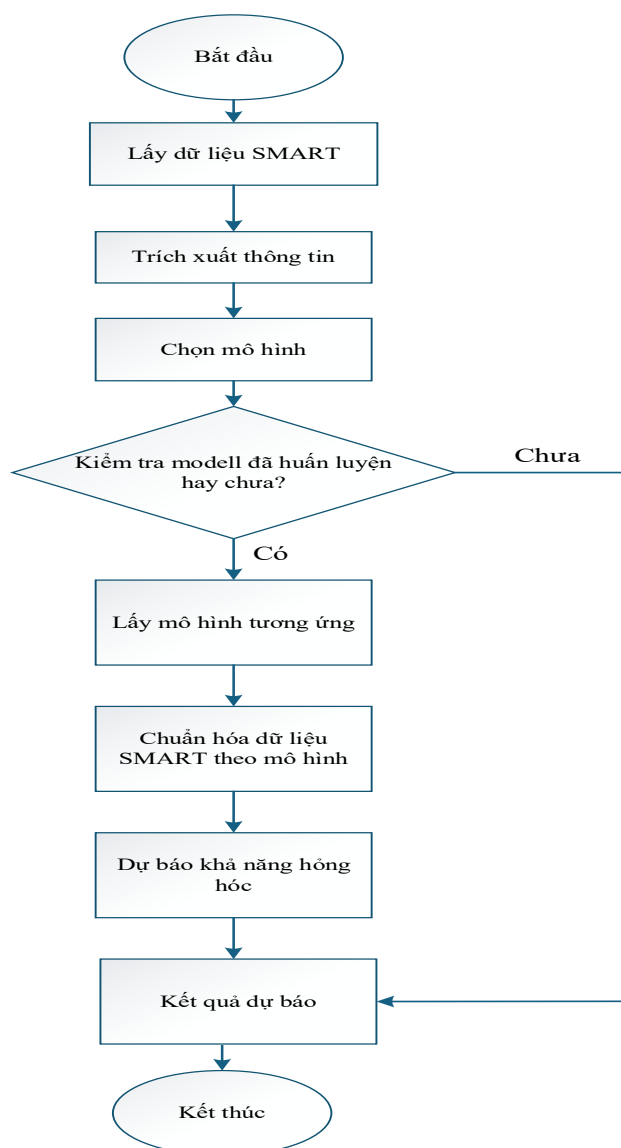
Hình 2.5. Tiến trình huấn luyện mô hình RF và XGB

2.3.4. Tiến trình dự đoán, phát hiện hỏng hóc ổ cứng

Sử dụng mô hình đã huấn luyện để dự đoán kết quả khả năng hỏng hóc của một ổ đĩa với dữ liệu SMART thu thập được trong thực tế. Các bước gồm: Đọc dữ liệu SMART (do Agent thu thập đầy về, Chuẩn hóa dữ liệu, Chọn mô hình dự báo, Dự báo, Hiển thị kết quả.



Hình 2.6. Tiến trình dự đoán phần trăm hỏng hóc ổ đĩa cứng



Hình 2.7. Lưu đồ thuật toán dự đoán hỏng hóc ổ đĩa cứng

2.3.5. Kết quả dự báo

Sau khi huấn luyện xong, mỗi model ổ đĩa cứng được lưu thành 01 file .pkl tương ứng với mô hình RF hoặc XGB. Với phương pháp tiếp cận, mỗi model ổ cứng sẽ có tập đặc trưng khác nhau:

- **Model ST12000NM001G**: 18 đặc trưng

```
['smart_1_raw', 'smart_4_raw', 'smart_5_raw', 'smart_7_raw', 'smart_9_raw',  
'smart_12_raw', 'smart_187_raw', 'smart_188_raw', 'smart_190_raw', 'smart_192_raw',  
'smart_193_raw', 'smart_194_raw', 'smart_197_raw', 'smart_198_raw', 'smart_199_raw',  
'smart_240_raw', 'smart_241_raw', 'smart_242_raw']
```

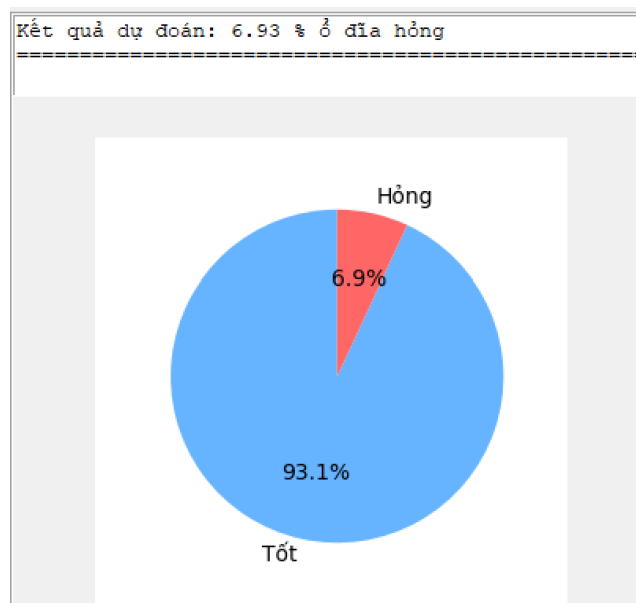
- **ST8000NM0055**: 20 đặc trưng

```
['smart_1_raw', 'smart_4_raw', 'smart_5_raw', 'smart_7_raw', 'smart_9_raw',  
'smart_12_raw', 'smart_187_raw', 'smart_188_raw', 'smart_190_raw', 'smart_191_raw',  
'smart_192_raw', 'smart_193_raw', 'smart_194_raw', 'smart_195_raw', 'smart_197_raw',  
'smart_198_raw', 'smart_199_raw', 'smart_240_raw', 'smart_241_raw', 'smart_242_raw']
```

- **ST4000DM000**: 21 đặc trưng

```
['smart_1_raw', 'smart_4_raw', 'smart_5_raw', 'smart_7_raw', 'smart_9_raw', 'smart_12_raw',  
'smart_183_raw', 'smart_184_raw', 'smart_187_raw', 'smart_188_raw', 'smart_189_raw',  
'smart_190_raw', 'smart_192_raw', 'smart_193_raw', 'smart_194_raw', 'smart_197_raw',  
'smart_198_raw', 'smart_199_raw', 'smart_240_raw', 'smart_241_raw', 'smart_242_raw']
```

Kết quả của nghiên cứu dự báo khả năng hỏng của ổ đĩa cứng theo tỉ lệ phần trăm hỏng và tỉ lệ phần trăm không hỏng



Hình 2.8: Kết quả dự báo tỉ lệ hỏng hóc của ổ đĩa cứng

2.4. Phương pháp đánh giá mô hình

Trong nghiên cứu, học viên chia tập dữ liệu SMART thành 02 thành phần:

- Dữ liệu huấn luyện (train): Chiếm tỉ lệ 70% hoặc 80%
- Dữ liệu kiểm thử (test) để đánh giá mô hình: Dữ liệu còn lại (chiếm 20% đến 30%).

Để đánh giá hiệu quả của mô hình, học viên đánh giá trên tập các giá trị sau:

- Tính chính xác (Accuracy) = (Số mẫu dự đoán đúng) / (Tổng số mẫu)
- Độ chính xác (Precision) = Tỉ lệ mẫu dự đoán là “hồng” thực sự bị hồng (giảm báo

động giả). $Precision = TP / (TP + FP)$. Trong đó:

+ TP: True Positive (dự đoán đúng hồng)

+ FP: False Positive (báo sai là hồng $\rightarrow$ ô tốt mà báo hồng)

- Độ bao phủ (Recall): Tỉ lệ ô hồng thực sự được mô hình phát hiện đúng (giảm bỏ sót ô hồng). $Recall = TP / (TP + FN)$. Trong đó:

+ TP: True Positive (dự đoán đúng hồng)

+ FN: False Negative (bỏ sót $\rightarrow$ ô hồng mà báo tốt)

- Trung bình điều hòa giữa Precision và Recall (F1-score): F1-score là chỉ số lý tưởng để đánh giá mô hình phân loại mất cân bằng, giúp cân bằng giữa “phát hiện đúng” và “tránh báo động sai”.

$$F1\text{-score} = 2 \times ((Precision \times Recall) / (Precision + Recall))$$

Bảng 2.3. Mối liên hệ giữa các giá trị đánh giá mô hình

Tình huống thực tế	Precision	Recall	F1-score	Giải thích
Mô hình báo đúng hồng và không hồng	Cao	Cao	Cao	Dự đoán tốt, cân bằng
Mô hình báo sai nhiều ô tốt là hồng	Cao	Thấp	Trung bình/thấp	Báo đúng thì đúng, nhưng bỏ sót nhiều ô hồng
Mô hình báo tất cả đều hồng	Thấp	Cao	Trung bình/thấp	Không bỏ sót nhưng báo sai quá nhiều

Chương 3: TRIỂN KHAI THỬ NGHIỆM, ĐÁNH GIÁ

Nội dung chương 3 trình bày về việc triển khai thử nghiệm hệ thống phát hiện, dự báo hồng hóc ổ cứng trong mạng quân sự đã xây dựng. Các nội dung chính gồm: thiết lập môi trường thử nghiệm, hệ Agent thu thập dữ liệu SMART và cài đặt phần mềm; Triển khai thử nghiệm hệ thống với tập dữ liệu Backblaze và tập dữ liệu do học viên thu thập thực tế tại đơn vị công tác. Các kết quả thử nghiệm đạt được có độ chính xác cao, hệ thống hoạt động ổn định.

3.1. Thiết lập môi trường thử nghiệm

3.1.1. Môi trường phát triển phần mềm hệ thống

- Ngôn ngữ và thư viện lập trình: Sử dụng ngôn ngữ Python 3.12.0
- Thư viện học máy gồm: sklearn, xgboost, imbalanced-learn, shap
- Thư viện xử lý dữ liệu: pandas, numpy, ..
- Thư viện trực quan hóa: matplotlib, seaborn, ...
- IDE sử dụng: Jupyter Notebook, VSCode

Phần mềm thử nghiệm được tạo với một Project gồm các thành phần chính như sau:

+ data_cleaning: Có chức năng chuẩn hóa dữ liệu SMART trước khi đưa vào mô hình huấn luyện.

+ dataset: chứa 03 tập dữ liệu: Dữ liệu thu thập từ Backblaze (chưa được chuẩn hóa), dữ liệu để huấn luyện mô hình (đã chuẩn hóa), dữ liệu thực tế thu thập được (dữ liệu dự báo).

+ models: Lưu mô hình đã huấn luyện gồm mô hình RF và XGB.

+ src: Chứa các hàm để huấn luyện, dự báo, kết xuất dữ liệu dự báo (giao diện)

+ requiremnets.txt: File chứa các thư viện cần phải cài đặt cho chương trình.

3.1.2. Hệ Agent thu thập dữ liệu SMART

- Dữ liệu huấn luyện: Sử dụng dữ liệu Backblaze từ năm 2022 đến nay.
- Dữ liệu thực tế tại đơn vị công tác: Agent tự động lấy dữ liệu SMART qua công cụ smartctl, lưu định kỳ 1 ngày/lần. Thử nghiệm với 02 máy chủ Windows, 01 Linux và 03 máy trạm tại đơn vị công tác.

3.1.3. Triển khai các thuật toán học máy

3.1.3.1. Triển khai thuật toán Random Forest

Thuật toán được khởi tạo với bộ tham số:

- $n_estimators=2000$: Số lượng cây quyết định trong rừng.
- $min_samples_split=5$: Một nút sẽ được chia nếu có ít nhất 5 mẫu .

- *min_samples_leaf*=4: Mỗi nút lá phải chứa ít nhất 4 mẫu.
- *max_features*='sqrt': Ở mỗi nút chia, chỉ xem xét $\sqrt{(\text{số đặc trưng})}$.
- *max_depth*=10: Giới hạn độ sâu của cây để kiểm soát độ phức tạp, giảm overfitting.
- *criterion*='entropy': Hàm đo mức độ thu được thông tin khi chia.
- *bootstrap*=True: Lấy mẫu với lặp lại để huấn luyện từng cây

Ngoài ra, sử dụng hàm `RandomizedSearchCV()` để tối ưu các siêu tham số cho mô hình. Các thông số chính của hàm:

- *n_iter*=100: Thử ngẫu nhiên 100 tổ hợp siêu tham số từ `random_grid` (giảm thời gian so với `GridSearchCV`).
- *cv* =3: Chia dữ liệu thành 3 phần (k-fold cross validation) để đánh giá ổn định mỗi cấu hình.
- *scoring*=['f1', "accuracy"]: Đánh giá mô hình theo cả F1-score và Accuracy.

Ngoài ra để tối ưu các bộ tham số sử dụng trong thuật toán Random Forest, đề án đã sử dụng các kỹ thuật ***RandomizedSearchCV*** để tự động tìm ra cấu hình tham số tốt nhất và tối ưu hiệu suất thuật toán Random Forest. Kỹ thuật ***Random Forest kết hợp Sliding Time Window (STW) và cơ chế bỏ phiếu chọn lọc (Part-voting)***: được áp dụng để tăng độ ổn định và tính chính xác trong phát hiện và dự báo lỗi ổ cứng.

3.1.3.2. Triển khai thuật toán XGBoost Classifier

Khởi tạo bộ tham số cho thuật toán XGBoost Classifier như sau:

- *learning_rate*=0.2: Tốc độ học
- *n_estimators*=1251: Số cây được huấn luyện
- *max_depth* = 3: Độ sâu tối đa của mỗi cây – kiểm soát độ phức tạp.
- *min_child_weight* =3: Số lượng mẫu tối thiểu trong một nút để chia tiếp.
- *gamma* = 0.1: Mức tăng lợi ích tối thiểu để chia cây.
- *subsample* = 0.8: Tỷ lệ mẫu được chọn ngẫu nhiên để huấn luyện mỗi cây.
- *colsample_bytree* = 0.8: Tỷ lệ số đặc trưng được chọn ngẫu nhiên cho mỗi cây.
- *objective* = 'binary:logistic': Bài toán phân loại nhị phân, đầu ra là xác suất.
- *scale_pos_weight* = 1: Trọng số lớp dương – sử dụng khi dữ liệu mất cân bằng.

Ngoài ra, đề án sử dụng hàm `RandomizedSearchCV()` để tối ưu các siêu tham số cho mô hình XGB.

3.2. Kết quả thử nghiệm trên tập dữ liệu của Backblaze

3.2.1. Tập dữ liệu SMART của Backblaze

Trong nghiên cứu này, học viên sử dụng dữ liệu năm 2022 để huấn luyện mô hình với các tham số sau:

- Dung lượng (dữ liệu gốc): 26,3 GB (Dữ liệu trong 1 năm, tần suất 01 lần/ngày).
- Model ổ cứng: 71 model ổ cứng được thu thập. Số lượng mẫu thu thập một số model ổ cứng được nêu ở Bảng 3.1

Bảng 3.1. Tổng hợp mẫu dữ liệu một số model ổ cứng của Backblaze

Số lượng mẫu	Model ổ cứng	Số lượng mẫu tốt	Số mẫu hỏng
1682389	ST4000DM000	1682231	158
1826833	ST12000NM0008	1826734	99
3513488	TOSHIBA MG07ACA14TA	3513405	83
1323908	ST8000NM0055	1323832	76
1873090	ST16000NM001G	1873045	45
878386	ST8000DM002	878345	41
988602	ST14000NM001G	988568	34
1157857	ST12000NM001G	1157831	26
991452	HGST HUH721212ALN604	991429	23
107973	ST10000NM0086	107955	18
1210844	HGST HUH721212ALE604	1210830	14
140485	ST14000NM0138	140471	14
546519	TOSHIBA MG08ACA16TE	546506	13
116481	ST12000NM0007	116471	10
342851	HGST HMS5C4040ALE640	342843	8
22343	TOSHIBA MQ01ABF050	22336	7
23773	TOSHIBA MQ01ABF050M	23766	7

Số lượng mẫu	Model ổ cứng	Số lượng mẫu tốt	Số mẫu hỏng
1170975	HGST HMS5C4040BLE640	1170968	7
455051	TOSHIBA MG08ACA16TEY	455045	6
847187	WDC WUH721816ALE6L4	847181	6
20711	ST500LM030	20706	5
345028	TOSHIBA MG08ACA16TA	345023	5
81512	ST6000DX000	81511	1

3.2.2. Kết quả thử nghiệm với tập dữ liệu Backblaze

Kết quả thử nghiệm cho 7 loại ổ cứng phổ biến được đánh giá dựa theo 04 tiêu chí gồm: Accuracy, Precision, Recall, F1-score. Kết quả thử nghiệm hệ thống với tập dữ liệu Backblaze được thể hiện ở Bảng 3.2 với 2 thuật toán Random Forest và XGBoost.

Bảng 3.2. Kết quả thử nghiệm mô hình với dữ liệu Backblaze

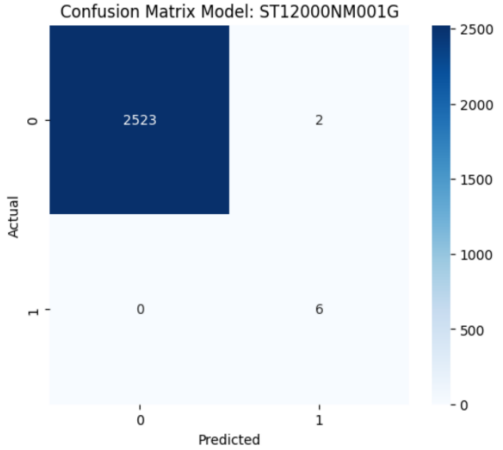
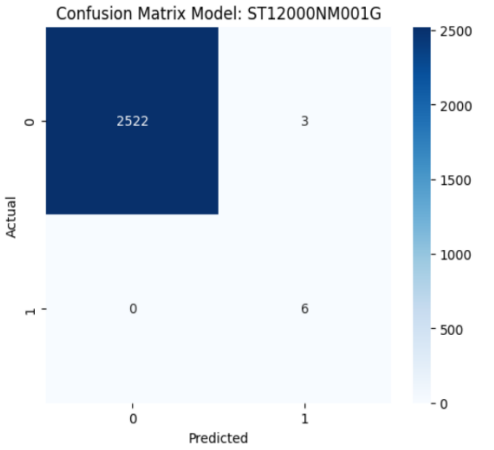
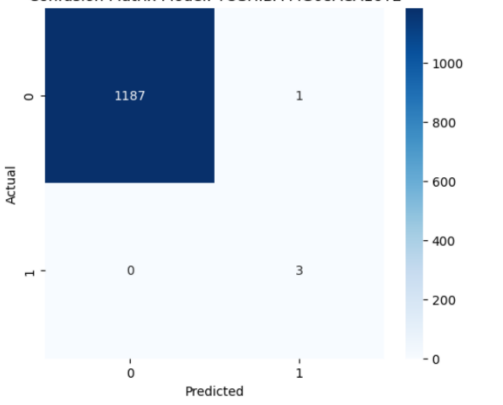
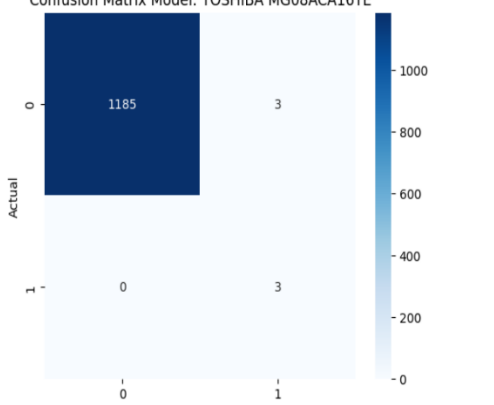
Model	Random Forest					XGBoost <i>Classifier</i>				
ST120 00NM 001G	- Results on test set: Accuracy: 0.9992097984986171 Scores:					- Results on test set: Accuracy: 0.9988146977479258 Scores:				
		precision	recall	f1-score	support		precision	recall	f1-score	support
	0	1.00	1.00	1.00	2525	0	1.00	1.00	1.00	2525
	1	0.75	1.00	0.86	6	1	0.67	1.00	0.80	6
	accuracy			1.00	2531	accuracy			1.00	2531
	macro avg	0.88	1.00	0.93	2531	macro avg	0.83	1.00	0.90	2531
	weighted avg	1.00	1.00	1.00	2531	weighted avg	1.00	1.00	1.00	2531
ST120 00NM 0008	- Results on test set: Accuracy: 0.9959849435382685 Scores:					- Results on test set: Accuracy: 0.9962358845671268 Scores:				
		precision	recall	f1-score	support		precision	recall	f1-score	support
	0	1.00	1.00	1.00	3965	0	1.00	1.00	1.00	3965
	1	0.56	1.00	0.71	20	1	0.57	1.00	0.73	20
	accuracy			1.00	3985	accuracy			1.00	3985
	macro avg	0.78	1.00	0.86	3985	macro avg	0.79	1.00	0.86	3985
	weighted avg	1.00	1.00	1.00	3985	weighted avg	1.00	1.00	1.00	3985

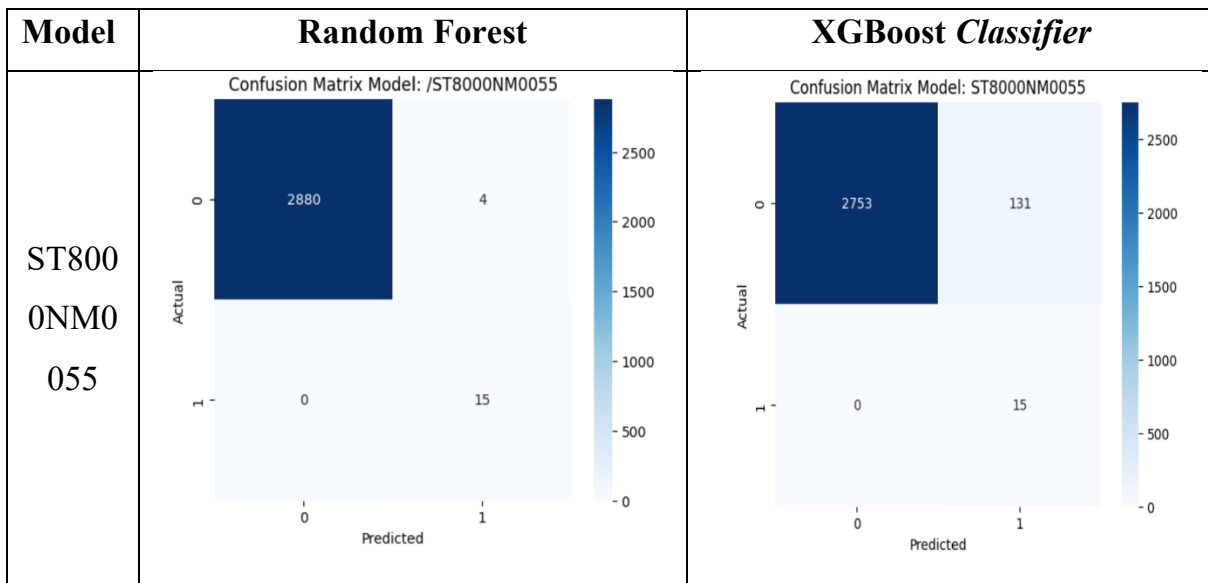
Model	Random Forest	XGBoost Classifier																																																												
TOSH IBA MG08 ACA1 6TE	- Results on test set: Accuracy: 0.9991603694374476 Scores: <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>1.00</td><td>1.00</td><td>1.00</td><td>1188</td></tr><tr><td>1</td><td>0.75</td><td>1.00</td><td>0.86</td><td>3</td></tr><tr><td>accuracy</td><td></td><td></td><td>1.00</td><td>1191</td></tr><tr><td>macro avg</td><td>0.88</td><td>1.00</td><td>0.93</td><td>1191</td></tr><tr><td>weighted avg</td><td>1.00</td><td>1.00</td><td>1.00</td><td>1191</td></tr></tbody></table>		precision	recall	f1-score	support	0	1.00	1.00	1.00	1188	1	0.75	1.00	0.86	3	accuracy			1.00	1191	macro avg	0.88	1.00	0.93	1191	weighted avg	1.00	1.00	1.00	1191	- Results on test set: Accuracy: 0.9974811083123426 Scores: <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>1.00</td><td>1.00</td><td>1.00</td><td>1188</td></tr><tr><td>1</td><td>0.50</td><td>1.00</td><td>0.67</td><td>3</td></tr><tr><td>accuracy</td><td></td><td></td><td>1.00</td><td>1191</td></tr><tr><td>macro avg</td><td>0.75</td><td>1.00</td><td>0.83</td><td>1191</td></tr><tr><td>weighted avg</td><td>1.00</td><td>1.00</td><td>1.00</td><td>1191</td></tr></tbody></table>		precision	recall	f1-score	support	0	1.00	1.00	1.00	1188	1	0.50	1.00	0.67	3	accuracy			1.00	1191	macro avg	0.75	1.00	0.83	1191	weighted avg	1.00	1.00	1.00	1191
	precision	recall	f1-score	support																																																										
0	1.00	1.00	1.00	1188																																																										
1	0.75	1.00	0.86	3																																																										
accuracy			1.00	1191																																																										
macro avg	0.88	1.00	0.93	1191																																																										
weighted avg	1.00	1.00	1.00	1191																																																										
	precision	recall	f1-score	support																																																										
0	1.00	1.00	1.00	1188																																																										
1	0.50	1.00	0.67	3																																																										
accuracy			1.00	1191																																																										
macro avg	0.75	1.00	0.83	1191																																																										
weighted avg	1.00	1.00	1.00	1191																																																										
ST800 0NM0 055	- Results on test set: Accuracy: 0.9986202138668506 Scores: <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>1.00</td><td>1.00</td><td>1.00</td><td>2884</td></tr><tr><td>1</td><td>0.79</td><td>1.00</td><td>0.88</td><td>15</td></tr><tr><td>accuracy</td><td></td><td></td><td>1.00</td><td>2899</td></tr><tr><td>macro avg</td><td>0.89</td><td>1.00</td><td>0.94</td><td>2899</td></tr><tr><td>weighted avg</td><td>1.00</td><td>1.00</td><td>1.00</td><td>2899</td></tr></tbody></table>		precision	recall	f1-score	support	0	1.00	1.00	1.00	2884	1	0.79	1.00	0.88	15	accuracy			1.00	2899	macro avg	0.89	1.00	0.94	2899	weighted avg	1.00	1.00	1.00	2899	- Results on test set: Accuracy: 0.9548120041393584 Scores: <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>1.00</td><td>0.95</td><td>0.98</td><td>2884</td></tr><tr><td>1</td><td>0.10</td><td>1.00</td><td>0.19</td><td>15</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.95</td><td>2899</td></tr><tr><td>macro avg</td><td>0.55</td><td>0.98</td><td>0.58</td><td>2899</td></tr><tr><td>weighted avg</td><td>1.00</td><td>0.95</td><td>0.97</td><td>2899</td></tr></tbody></table>		precision	recall	f1-score	support	0	1.00	0.95	0.98	2884	1	0.10	1.00	0.19	15	accuracy			0.95	2899	macro avg	0.55	0.98	0.58	2899	weighted avg	1.00	0.95	0.97	2899
	precision	recall	f1-score	support																																																										
0	1.00	1.00	1.00	2884																																																										
1	0.79	1.00	0.88	15																																																										
accuracy			1.00	2899																																																										
macro avg	0.89	1.00	0.94	2899																																																										
weighted avg	1.00	1.00	1.00	2899																																																										
	precision	recall	f1-score	support																																																										
0	1.00	0.95	0.98	2884																																																										
1	0.10	1.00	0.19	15																																																										
accuracy			0.95	2899																																																										
macro avg	0.55	0.98	0.58	2899																																																										
weighted avg	1.00	0.95	0.97	2899																																																										
ST100 00NM 0086	- Results on test set: Accuracy: 1.0 Scores: <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>1.00</td><td>1.00</td><td>1.00</td><td>235</td></tr><tr><td>1</td><td>1.00</td><td>1.00</td><td>1.00</td><td>4</td></tr><tr><td>accuracy</td><td></td><td></td><td>1.00</td><td>239</td></tr><tr><td>macro avg</td><td>1.00</td><td>1.00</td><td>1.00</td><td>239</td></tr><tr><td>weighted avg</td><td>1.00</td><td>1.00</td><td>1.00</td><td>239</td></tr></tbody></table>		precision	recall	f1-score	support	0	1.00	1.00	1.00	235	1	1.00	1.00	1.00	4	accuracy			1.00	239	macro avg	1.00	1.00	1.00	239	weighted avg	1.00	1.00	1.00	239	- Results on test set: Accuracy: 1.0 Scores: <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>1.00</td><td>1.00</td><td>1.00</td><td>235</td></tr><tr><td>1</td><td>1.00</td><td>1.00</td><td>1.00</td><td>4</td></tr><tr><td>accuracy</td><td></td><td></td><td>1.00</td><td>239</td></tr><tr><td>macro avg</td><td>1.00</td><td>1.00</td><td>1.00</td><td>239</td></tr><tr><td>weighted avg</td><td>1.00</td><td>1.00</td><td>1.00</td><td>239</td></tr></tbody></table>		precision	recall	f1-score	support	0	1.00	1.00	1.00	235	1	1.00	1.00	1.00	4	accuracy			1.00	239	macro avg	1.00	1.00	1.00	239	weighted avg	1.00	1.00	1.00	239
	precision	recall	f1-score	support																																																										
0	1.00	1.00	1.00	235																																																										
1	1.00	1.00	1.00	4																																																										
accuracy			1.00	239																																																										
macro avg	1.00	1.00	1.00	239																																																										
weighted avg	1.00	1.00	1.00	239																																																										
	precision	recall	f1-score	support																																																										
0	1.00	1.00	1.00	235																																																										
1	1.00	1.00	1.00	4																																																										
accuracy			1.00	239																																																										
macro avg	1.00	1.00	1.00	239																																																										
weighted avg	1.00	1.00	1.00	239																																																										
ST140 00NM 0138	- Results on test set: Accuracy: 0.993485342019544 Scores: <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>1.00</td><td>0.99</td><td>1.00</td><td>304</td></tr><tr><td>1</td><td>0.60</td><td>1.00</td><td>0.75</td><td>3</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.99</td><td>307</td></tr><tr><td>macro avg</td><td>0.80</td><td>1.00</td><td>0.87</td><td>307</td></tr><tr><td>weighted avg</td><td>1.00</td><td>0.99</td><td>0.99</td><td>307</td></tr></tbody></table>		precision	recall	f1-score	support	0	1.00	0.99	1.00	304	1	0.60	1.00	0.75	3	accuracy			0.99	307	macro avg	0.80	1.00	0.87	307	weighted avg	1.00	0.99	0.99	307	- Results on test set: Accuracy: 0.9804560260586319 Scores: <table><thead><tr><th></th><th>precision</th><th>recall</th><th>f1-score</th><th>support</th></tr></thead><tbody><tr><td>0</td><td>1.00</td><td>0.98</td><td>0.99</td><td>304</td></tr><tr><td>1</td><td>0.33</td><td>1.00</td><td>0.50</td><td>3</td></tr><tr><td>accuracy</td><td></td><td></td><td>0.98</td><td>307</td></tr><tr><td>macro avg</td><td>0.67</td><td>0.99</td><td>0.75</td><td>307</td></tr><tr><td>weighted avg</td><td>0.99</td><td>0.98</td><td>0.99</td><td>307</td></tr></tbody></table>		precision	recall	f1-score	support	0	1.00	0.98	0.99	304	1	0.33	1.00	0.50	3	accuracy			0.98	307	macro avg	0.67	0.99	0.75	307	weighted avg	0.99	0.98	0.99	307
	precision	recall	f1-score	support																																																										
0	1.00	0.99	1.00	304																																																										
1	0.60	1.00	0.75	3																																																										
accuracy			0.99	307																																																										
macro avg	0.80	1.00	0.87	307																																																										
weighted avg	1.00	0.99	0.99	307																																																										
	precision	recall	f1-score	support																																																										
0	1.00	0.98	0.99	304																																																										
1	0.33	1.00	0.50	3																																																										
accuracy			0.98	307																																																										
macro avg	0.67	0.99	0.75	307																																																										
weighted avg	0.99	0.98	0.99	307																																																										

Model	Random Forest					XGBoost <i>Classifier</i>				
ST800 ODM0 02	- Results on test set: Accuracy: 0.9963427377220481 Scores:					- Results on test set: Accuracy: 0.9937304075235109 Scores:				
		precision	recall	f1-score	support		precision	recall	f1-score	support
	0	1.00	1.00	1.00	1905	0	1.00	0.99	1.00	1905
	1	0.56	1.00	0.72	9	1	0.43	1.00	0.60	9
	accuracy			1.00	1914	accuracy			0.99	1914
	macro avg	0.78	1.00	0.86	1914	macro avg	0.71	1.00	0.80	1914
	weighted avg	1.00	1.00	1.00	1914	weighted avg	1.00	0.99	0.99	1914

Bảng 3.3 thể hiện kết quả đánh giá mô hình bằng ma trận Confusion với 2 thuật toán Random Forest và XGBoost cho 3 loại ổ cứng.

Bảng 3.3. Kết quả đánh giá mô hình bằng ma trận Confusion

Model	Random Forest	XGBoost Classifier
ST120 00NM 001G	Confusion Matrix Model: ST12000NM001G 	Confusion Matrix Model: ST12000NM001G 
TOSHI BA MG08 ACA1 6TE	Confusion Matrix Model: TOSHIBA MG08ACA16TE 	Confusion Matrix Model: TOSHIBA MG08ACA16TE 



Bảng 3.4. Kết quả đánh giá mô hình dựa trên Precision, Recall, F1-score

Model ổ cứng	Phân lớp ổ cứng	Precision		Recall		F1-score	
		RF	XGB	RF	XGB	RF	XGB
ST12000NM001G	Tốt (0)	1,00	1,00	1,00	1,00	1,00	1,00
	Hỏng (1)	0,75	0,67	1,00	1,00	0,86	0,80
ST12000NM0008	Tốt (0)	1,00	1,00	1,00	1,00	1,00	1,00
	Hỏng (1)	0,56	0,56	1,00	1,00	0,71	0,73
TOSHIBA MG08ACA16TE	Tốt (0)	1,00	1,00	1,00	1,00	1,00	1,00
	Hỏng (1)	0,75	0,50	1,00	1,00	0,86	0,67
ST8000NM0055	Tốt (0)	1,00	1,00	1,00	1,00	0,95	0,98
	Hỏng (1)	0,79	0,10	1,00	1,00	0,88	0,19
ST10000NM0086	Tốt (0)	1,00	1,00	1,00	1,00	1,00	1,00
	Hỏng (1)	1,00	1,00	1,00	1,00	1,00	1,00
ST14000NM0138	Tốt (0)	1,00	1,00	0,99	0,98	1,00	0,99
	Hỏng (1)	0,60	0,33	1,00	1,00	0,75	0,50
ST8000DM002	Tốt (0)	1,00	1,00	1,00	0,99	1,00	1,00
	Hỏng (1)	0,56	0,43	1,00	1,00	0,72	0,60

Từ thử nghiệm trên tập dữ liệu SMART của Backblaze, các kết quả trên Bảng 3.2, 3.3 và Bảng 3.4 cho thấy: Mô hình sử dụng trong nghiên cứu không bỏ sót ổ cứng lỗi (tỷ lệ recall ~ 100%), tỉ lệ cảnh báo giả ở mức chấp nhận được (ma trận nhầm lẫn) cho hệ thống ưu tiên cảnh báo sớm (không bỏ sót ổ hỏng).

3.3. Thử nghiệm với dữ liệu thu thập thực tế tại đơn vị công tác

Thực nghiệm với dữ liệu SMART thực tế tại đơn vị:

Dữ liệu SMART thu thập tại đơn vị được thể hiện ở Bảng 3.4, các trường dữ liệu SMART được thể hiện ở Hình 3.2 (Model ổ cứng ST4000DM000).

Bảng 3.4. Dữ liệu SMART thu thập tại đơn vị công tác

Loại máy	Model ổ cứng	Chỉ số SMART	Số mẫu hỏng
Máy trạm	ST4000DM000	19	0

Loại máy	Model ổ cứng	Chỉ số SMART	Số mẫu hỏng
Máy trạm	KBG4AZNV512G KIOXIA	18	0
Máy trạm	SANDISK Z400S 2.5 7MM 256GB	17	0
Máy chủ	HGST HTS545050A7E680	23	0
Máy chủ	ST1000DM010-2EP102	23	0
Máy trạm	TOSHIBA DT01ACA050	17	0

```

=== START OF INFORMATION SECTION ===
Model Number:          ST4000DM000
Serial Number:         AYD7N017811407K2Y
Firmware Version:      61001141
PCI Vendor/Subsystem ID: 0x1c5c
IEEE OUI Identifier:   0xace42e
Controller ID:         1
NVMe Version:          1.4
Number of Namespaces:  1
Namespace 1 Size/Capacity: 1,024,209,543,168 [1.02 TB]
Namespace 1 Formatted LBA Size: 512
Namespace 1 IEEE EUI-64: ace42e 004aa44151
Local Time is:         Mon May 12 14:24:55 2025 SEAST
Firmware Updates (0x16): 3 Slots, no Reset required
Optional Admin Commands (0x0017): Security Format Frmw_DL Self_Test
Optional NVM Commands (0x00df): Comp Wr_Unc DS_Mngmt Wr_Zero Sav/Sel_Feat Timestmp Verify
Log Page Attributes (0x1e): Cmd_Eff_Lg Ext_Get_Lg Telmtry_Lg Pers_Ev_Lg
Maximum Data Transfer Size: 64 Pages
Warning Comp. Temp. Threshold: 83 Celsius
Critical Comp. Temp. Threshold: 85 Celsius


Supported Power States
St Op   Max Active   Idle   RL RT WL WT  Ent_Lat  Ex_Lat
0 +     7.50W    -       -     0 0 0 0       5       305
1 +     3.900W  -       -     1 1 1 1      30       330
2 +     1.500W  -       -     2 2 2 2     100       400
3 -     0.050W  -       -     3 3 3 3     500      1500
4 -     0.005W  -       -     4 4 4 4    1000      9000


Supported LBA Sizes (NSID 0x1)
Id Fmt  Data  Metadt  Rel_Perf
0 +     512     0       0
1 -    4096     0       0


=== START OF SMART DATA SECTION ===
SMART overall-health self-assessment test result: PASSED


SMART/Health Information (NVMe Log 0x02)
Critical Warning:          0x00
Temperature:              29 Celsius
Available Spare:           100%
Available Spare Threshold: 50%
Percentage Used:           0%
Data Units Read:          4,787,659 [2.45 TB]
Data Units Written:        4,484,998 [2.29 TB]
Host Read Commands:       34,309,418
Host Write Commands:      28,698,344
Controller Busy Time:     606
Power Cycles:              137
Power On Hours:            198
Unsafe Shutdowns:         59
Media and Data Integrity Errors: 0
Error Information Log Entries: 0
Warning Comp. Temperature Time: 0
Critical Comp. Temperature Time: 0
Temperature Sensor 1:     29 Celsius
Temperature Sensor 2:     30 Celsius

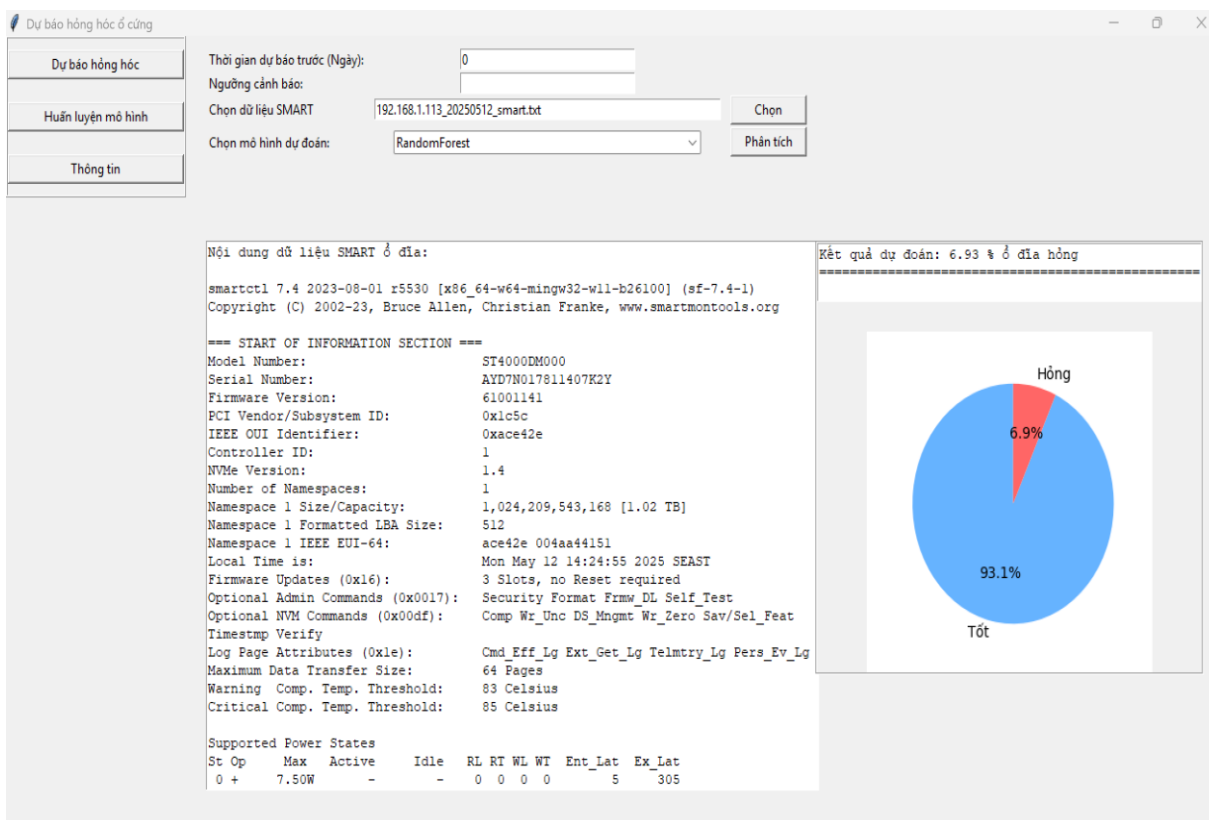
```

Hình 3.2. Dữ liệu SMART thu thập được tại đơn vị (Model ST4000DM000)

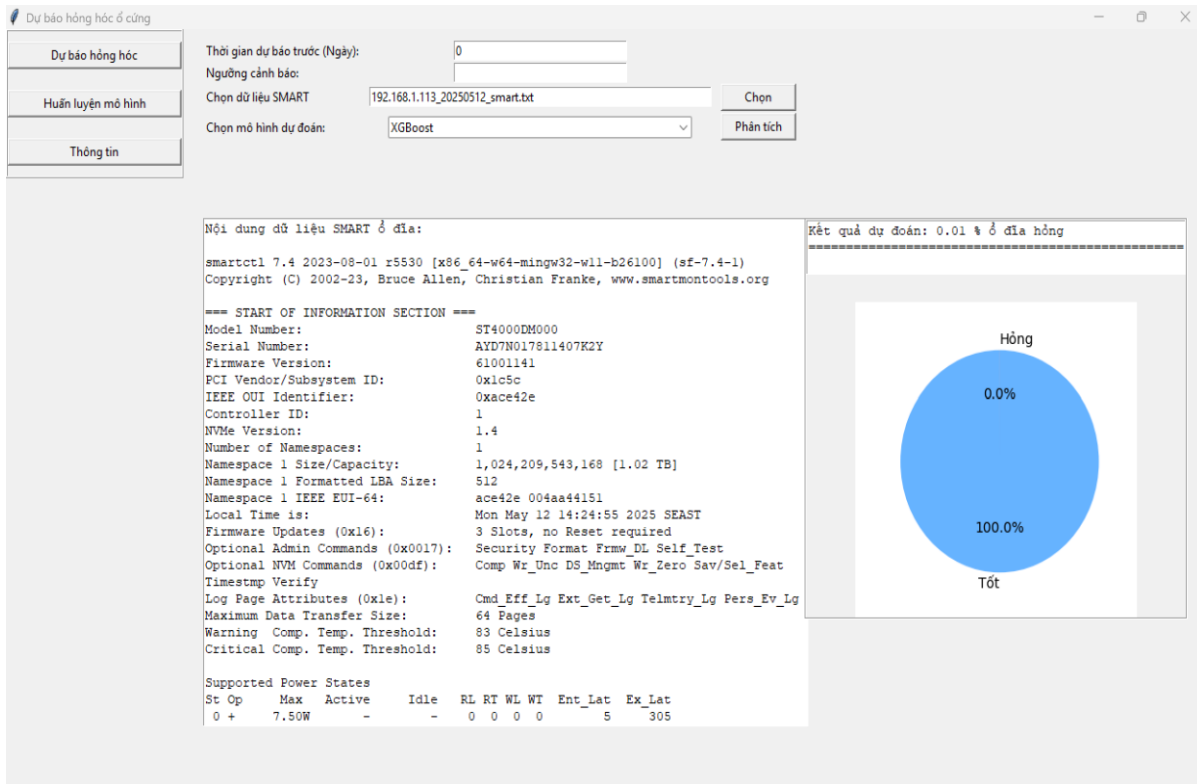
- Kết quả dự báo hỏng hóc với model ổ đĩa đã được huấn luyện được thể hiện ở hình 3.2 với thuật toán Random Forest, Hình 3.3 với thuật toán XGB. (Model ổ đĩa ST4000DM000, thu thập từ máy trạm có IP 192.168.1.113). Các tham số được thiết lập:

+ Sử dụng thuật toán RF hoặc XGB.
+ Không thiết lập dự báo khoảng thời gian (STW): Số ngày dự báo trước = 0, Ngưỡng cảnh báo: Không thiết lập.

+ Kết quả dự báo khả năng hỏng hóc của ổ cứng tạo thời điểm thu thập: Với dữ liệu SMART của ổ cứng thu thập được hệ thống đưa ra dự báo hỏng của ổ đĩa là: 6,93% với mô hình RF và 0,01% với mô hình XGB.



Hình 3.2. Kết quả dự báo hỏng hóc với dữ liệu thực tế với mô hình RF



Hình 3.3. Kết quả dự báo hỏng ổ cứng với dữ liệu thực tế với mô hình XGB

- Với model ổ cứng chưa được huấn luyện: Các tham số hệ thống được cấu hình tương tự như đối với model ST4000DM000 đã được huấn luyện ở trên. Với trường hợp này hệ thống đưa ra cảnh báo “*Model ổ cứng chưa được huấn luyện để dự đoán hỏng ổ cứng*” (Hình 3.4).



Hình 3.4. Kết quả dự báo hỏng ổ cứng với model chưa được huấn luyện

KẾT LUẬN

1. Kết quả đạt được của đề án

Trong bối cảnh lượng dữ liệu ngày càng nhiều được lưu trữ trên các ổ cứng trên mạng hay qua các dịch vụ lưu trữ đám mây, việc giám sát, phát hiện và dự báo lỗi ổ cứng là một nhu cầu cấp thiết và đang ngày càng được quan tâm hơn. Vấn đề này càng quan trọng đối với các mạng máy tính đóng như mạng chuyên dụng quân sự, điển hình là mạng máy tính của Binh chủng Thông tin liên lạc thuộc Bộ Quốc phòng, nơi học viên đang công tác.

Đã có các giải pháp truyền thống để giám sát và phát hiện lỗi ổ cứng như: sử dụng các công cụ tiện ích của hệ điều hành hoặc một số phần mềm giám sát hoạt động máy tính. Tuy nhiên, các giải pháp này hầu hết vẫn chỉ thủ công, chưa chính xác, chưa có khả năng phát hiện sớm sự cố lỗi của các đĩa cứng đang hoạt động cũng như khả năng dự báo hỏng hóc theo thời gian. Áp dụng các kỹ thuật học máy, học sâu tạo ra triển vọng mới trong phát hiện và dự báo hỏng hóc của các trang thiết bị nói chung và các đĩa cứng nói riêng, tạo ra khả năng tự động hóa và giúp nâng cao tính ổn định, độ chính xác. Tuy nhiên, vẫn chưa có một hệ thống như vậy được đưa vào ứng dụng trong thực tế, đặc biệt là đối với mạng chuyên dụng của quân sự.

Đề tài của đề án này là nghiên cứu, xây dựng một hệ thống phát hiện và dự báo lỗi ổ cứng hướng tới mục tiêu triển khai trong mạng máy tính quân sự của Binh chủng Thông tin liên lạc. Để xây dựng hệ thống, cần nghiên cứu nguồn dữ liệu lỗi ổ cứng với các thuộc tính SMART, các kỹ thuật học máy/ học sâu phù hợp cho tập dữ liệu, lựa chọn mô hình phù hợp cho mạng chuyên dụng quân sự tại đơn vị. Đây là những nội dung nghiên cứu có tính khoa học, tính cấp thiết và thực tế.

Các kết quả chính đã đạt được của đề án gồm:

- Nghiên cứu cơ sở lý thuyết về ổ cứng, cơ chế bảo vệ dữ liệu, bài toán phát hiện và dự báo lỗi ổ cứng thông qua công nghệ SMART và những vấn đề liên quan đến bài toán, cụ thể về cách thức phát hiện và dự báo lỗi ổ cứng theo kiểu truyền thống và theo hướng áp dụng kỹ thuật học máy/học sâu.

- Nghiên cứu và đề xuất một mô hình hệ thống phát hiện, dự báo hỏng hóc ổ cứng trong mạng quân sự với việc xây dựng và triển khai một hệ Agent thu thập dữ liệu SMART từ các ổ cứng, một hệ xử lý trung tâm áp dụng các thuật toán học máy gồm Random Forest và XGBoost và các kỹ thuật giúp lựa chọn tham số tối ưu như RandomizedSearchCV và kỹ thuật trợ giúp nâng cao độ chính xác như Sliding Time Window (STW) và cơ chế bỏ phiếu

chọn lọc (Part-voting). Đưa ra lưu đồ triển khai dự đoán hỏng hóc ổ cứng, đánh giá mô hình qua các bộ tham số Accuracy, Precision, Recall, F1-score, ma trận nhầm lẫn.

- Triển khai thiết lập môi trường thử nghiệm, cài đặt bộ phần mềm thu thập dữ liệu SMART của Backblaze và tập dữ liệu do học viên thu thập thực tế tại đơn vị công tác, triển khai thử nghiệm hệ thống với 2 tập dữ liệu trên. Các kết quả thử nghiệm cho thấy hệ thống với tập dữ liệu của Backblaze và với dữ liệu thực tế thu thập tại đơn vị công tác, Mô hình không bỏ sót ổ cứng lỗi (thông qua tỷ lệ recall xấp xỉ 100%), Tỷ lệ cảnh báo giả ở mức chấp nhận được với việc hệ thống ưu tiên cảnh báo sớm (không bỏ sót ổ cứng hỏng).

Tập dữ liệu thu thập từ Backblaze khá lớn bảo đảm độ tin cậy của thử nghiệm. Các kết quả đạt được của hệ thống với các mô hình sử dụng Random Forest và XGBoost đều cho kết quả tốt và phù hợp với bài toán phát hiện hỏng hóc ổ đĩa cứng trong thực tế.

2. Hướng phát triển tiếp theo

Tuy nhiên trong phạm vi nghiên cứu của đề án, còn một số hạn chế về mặt thời gian thực hiện, cũng như dữ liệu thực tế thu thập được. Để đề án được hoàn thiện và có thể triển khai trong thực tế, cần hoàn thiện một số nội dung sau:

- Nghiên cứu, mở rộng mô hình sang học sâu (deep learning), đặc biệt là các mô hình RNN hoặc Transformer để xử lý chuỗi dữ liệu SMART.

- Kết hợp dữ liệu SMART với log hệ thống (log hệ điều hành, ứng dụng) để tăng khả năng dự báo chính xác hơn.

- Xây dựng hệ thống có khả năng phát triển mô hình học liên tục (online learning) phù hợp với dữ liệu thực tế cập nhật liên tục.

- Hoàn thiện giao diện người dùng: Giao diện cho phép huấn luyện mô hình, giao diện kết xuất dữ liệu dự báo theo thời gian (tuần, tháng, quý, năm) phục vụ quản trị hệ thống (trung tâm dữ liệu, phòng máy chủ lớn) trong việc phát hiện sớm hỏng hóc liên quan đến ổ đĩa cứng.

- Triển khai thử nghiệm mô hình trong môi trường mạng quân sự và đánh giá hiệu quả lâu dài.