HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

GIÁP THỊ NGỌC BÍCH

NGHIÊN CỨU HỆ THỐNG PHÁT HIỆN TẤN CÔNG IDS TRONG MÔI TRƯỜNG IOT DỰA TRÊN HỌC MÁY

ĐỀ ÁN TỐT NGHIỆP THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

HÀ NỘI – 2025

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



GIÁP THỊ NGỌC BÍCH

NGHIÊN CỨU HỆ THỐNG PHÁT HIỆN TẤN CÔNG IDS TRONG MÔI TRƯỜNG IOT DỰA TRÊN HỌC MÁY

CHUYÊN NGÀNH: KỸ THUẬT VIỄN THÔNG
MÃ SỐ: 8.52.02.08

ĐỀ ÁN TỐT NGHIỆP THẠC SĨ KỸ THUẬT
(Theo định hướng ứng dụng)

NGƯỜI HƯỚNG DẪN KHOA HỌC:
PGS.TS. HOÀNG TRỌNG MINH

HÀ NỘI - 2025

LỜI CAM ĐOAN

Tôi cam đoan đây là công trình nghiên cứu của tôi.

Các số liệu, kết quả nêu trong đề án là trung thực và chưa từng được ai công bố trong bất kỳ công trình nào khác.

Tác giả đề án

Giáp Thị Ngọc Bích

LỜI CẢM ƠN

Em xin được bày tỏ sự tri ân chân thành đến toàn thể quý thầy cô và cán bộ của Học viện Công nghệ Bưu chính Viễn thông Hà Nội, đặc biệt là các giảng viên thuộc Khoa Kỹ Thuật Viễn Thông và Khoa Sau Đại học.

Đặc biệt, em xin gửi lời cảm ơn sâu sắc đến thầy PGS.TS.Hoàng Trọng Minhhh, người đã hướng dẫn em trong suốt quá trình thực hiện đề tài. Những góp ý sâu sắc, những chia sẻ về kinh nghiệm nghiên cứu, và sự tận tình chỉ bảo của thầy không chỉ giúp em hoàn thiện luận văn mà còn góp phần định hình phương pháp nghiên cứu khoa học của bản thân.

Bên cạnh đó, em xin gửi lời cảm ơn đến gia đình, người thân và bạn bè - những người đã luôn đồng hành, ủng hộ và động viên em trong suốt quá trình học tập và nghiên cứu. Sự hiện diện và những lời động viên kịp thời của họ là nguồn động lực to lớn giúp em vượt qua những thách thức và hoàn thành luận văn một cách tốt nhất.

Đề án này không thể tránh khỏi những hạn chế và thiếu sót, em chân thành mong nhận được những ý kiến đóng góp quý báu từ quý thầy cô, các nhà nghiên cứu và đồng nghiệp. Những góp ý này sẽ là cơ sở quan trọng để em tiếp tục hoàn thiện và nâng cao chất lượng nghiên cứu của mình, đồng thời mở rộng hiểu biết trong lĩnh vực chuyên môn.

Sinh viên thực hiện

Giáp Thị Ngọc Bích

MỤC LỤC

LỜI CAM ĐOAN ..... **i**
LỜI CẢM ƠN ..... **ii**
MỤC LỤC ..... **iii**
DANH MỤC CÁC THUẬT NGỮ, CHỮ VIẾT TẮT ..... **v**
DANH SÁCH BẢNG ..... **vi**
DANH SÁCH HÌNH VẼ ..... **vii**
MỞ ĐẦU **1**
CHƯƠNG 1: TỔNG QUAN VỀ HỆ THỐNG IDS TRONG IOT **2**
1.1. Tổng quan về IoT **2**
1.1.1.Lý thuyết về IoT (Internet Vạn Vật) 2
1.1.2. Kiến trúc IoT 4
1.1.3 Đặc trưng dữ liệu IoT 6
1.1.4.Một số công nghệ điển hình trong IoT 8
1.2. Tổng quan hệ thống IDS **11**
1.2.1. Lý thuyết về Hệ thống Phát hiện Xâm nhập (IDS) 11
1.2.2. Hệ thống IDS dựa trên học máy tại biên 15
1.3. Thách thức của các mô hình IDS trong môi trường IoT **16**
1.4. Kết Luận chương **17**
CHƯƠNG 2 - HỆ THỐNG PHÁT HIỆN TẤN CÔNG TRONG IOT DỰA TRÊN HỌC MÁY **18**
2.1. Hệ thống IDS dựa trên học máy cho thiết bị IoT tại biên **18**
2.1.1. Cơ sở khoa học của giải pháp đề xuất 18
2.1.2. Hệ thống IDS ứng dụng học máy đề xuất 19
2.2. Cơ sở lý thuyết của phương pháp phân cụm Fuzzy C-Mean **22**
2.3. Cơ sở lý thuyết của mô hình học máy **25**
2.3.1. Mô hình Decision tree 25
2.3.2. Mô hình Random forest 29
2.4. Tập dữ liệu CICIoT2023 ..... **31**

2.4.1. Nguồn gốc.....31
2.2.2. Cấu trúc tập dữ liệu.....33
2.2.3. Các loại tấn công trong tập dữ liệu.....35
2.2.4. Tập dữ liệu CICIoT2023 trong nghiên cứu và đề án.....38
2.2.5. Tiền xử lý dữ liệu 38
2.5. Kết luận chương 40
CHƯƠNG 3 - THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ CỦA MÔ HÌNH IDS CHO MÔI TRƯỜNG IOT.....42
3.1. Môi trường thực nghiệm.....42
3.1.1. Cấu hình của mô hình huấn luyện 42
3.1.2. Kịch bản thử nghiệm và đánh giá.....42
3.2. Kết quả đạt được 44
3.2.1. Các tiêu chí đánh giá hiệu năng 44
3.2.2. Đánh giá mô hình học máy khi không dùng phân cụm FCM 45
3.2.3. Đánh giá mô hình học máy khi sử dụng phương pháp phân cụm 46
3.3. Kết luận chương 3.....55
KẾT LUẬN VÀ KIẾN NGHỊ 57
DANH MỤC TÀI LIỆU THAM KHẢO 58

DANH MỤC CÁC THUẬT NGỮ, CHỮ VIẾT TẮT

Viết tắt	Tiếng Anh	Tiếng Việt
SSL	Secure Sockets Layer	Lớp ổ bảo mật
TCP	Transmission Control Protocol	Giao thức điều khiển truyền tải
UDP	User Datagram Protocol	Giao thức dữ liệu người dùng
XSS	Cross-site Scripting	Kịch bản chéo trang
API	Application Programming Interface	Giao diện lập trình ứng dụng
ARP	Address Resolution Protocol	Giao thức phân giải địa chỉ
CPU	Central Processing Unit	Bộ xử lý trung tâm
DoS / DDoS	Denial of Service / Distributed Denial of Service	Tấn công từ chối dịch vụ / tấn công từ chối dịch vụ phân tán
GPU	Graphics Processing Unit	Bộ xử lý đồ họa
HTTP	HyperText Transfer Protocol	Giao thức truyền tải siêu văn bản
IDS	Intrusion Detection System	Hệ thống phát hiện xâm nhập
IoT	Internet of Things	Internet vạn vật
MQTT	Message Queuing Telemetry Transport	Giao thức truyền thông hàng đợi

DANH SÁCH BẢNG

Bảng 2. 1 Phân bổ tấn công trong tập dữ liệu CICIOT2023 33
Bảng 2. 2 Các loại tấn công trong tập dữ liệu CICIOT2023 35
Bảng 2. 3 Gán nhãn các loại tấn công.....39
Bảng 3. 1 Độ chính xác của mô hình học máy trên tập dữ liệu CICIOT2023 45
Bảng 3. 2 Kết quả mô hình Decision tree kết hợp phân cụm FCM 47
Bảng 3. 3 Kết quả mô hình Random forest kết hợp phân cụm FCM.....48
Bảng 3. 4. So sánh hiệu suất mô hình học máy truyền thông và mô hình học máy kết hợp phân cụm FCM 49
Bảng 3. 5 Báo cáo phân loại theo lớp tấn công mô hình Decison Tree 50

DANH SÁCH HÌNH VẼ

Hình 1. 1 Kiến trúc hệ thống IoT 3 lớp.....4
Hình 1. 2 Phân loại mô hình học máy.....10
Hình 1. 3. Thành phần hệ thống IDS.....12
Hình 1. 4 Cách thức hoạt độn của SIDS.....13
Hình 2. 1 Mô hình hệ thống IDS đề xuất.....21
Hình 2. 2 Luồng dữ liệu hệ thuật toán FCM.....25
Hình 2. 3 Cấu trúc mô hình Decision tree.....27
Hình 2. 4 Cấu trúc mô hình Random forest.....29
Hình 2. 5 Phòng thí nghiệm CIC.....32

MỞ ĐẦU

Internet vạn vật đang dần dần trở thành xu hướng của thế giới. IoT xâm nhập vào trong mọi nghành nghề, trở thành thành phần then chốt trong cuộc sống. Sự phát triển nhanh chóng này giúp nâng cao tiện ích và chất lượng cuộc sống. Sự phát triển không ngừng của internet vạn vật (IoT) và điện toán biên (Edge Computing) đã tạo ra một lượng dữ liệu khổng lồ. Việc này tạo khó khăn cho việc quản lý dữ liệu, và tạo lỗ hổng bảo mật lớn cho hệ thống. Vì vậy việc phát triển và tối ưu hóa hệ thống phát hiện xâm nhập là vấn đề cấp thiết. Đặc biệt là các vấn đề bảo mật tại biên mạng, gần thiết bị người dùng. Tuy nhiên, thiết bị IoT có tài nguyên hạn chế, tính phân tán cao, việc thiết kế hệ thống IDS gặp thách thức rất lớn. Hệ thống IDS truyền thống trong môi trường này gặp nhiều khó khăn. Điện toán biên như một giải pháp giải quyết vấn đề trước mắt. Khi dữ liệu được xử lý ngay tại nguồn, giảm tải gánh năng cho mạng lưới mạng truyền thông, đồng thời cũng đưa ra quyết định, phản ứng nhanh chóng trong tình huống bất ngờ. Nhưng bên cạnh đó, tài nguyên phần cứng cũng lại là thách thức rất lớn cho hệ thống này. Xuất phát từ những vấn đề thực tế trên, đề án thực hiện với chủ đề " Nghiên cứu hệ thống phát hiện tấn công IDS ứng dụng học máy trong môi trường IoT". Mục tiêu chính của đề án là tìm ra phương pháp tối ưu cho hệ thống IDS nhẹ tại biên, vẫn duy trì được độ chính xác cao. Đề án cấu trúc gồm ba chương

Chương 1: Tổng quan về hệ thống IDS trong môi trường IoT

Chương 2: Hệ thống phát hiện tấn công trong IoT dựa trên học máy

Chươn 3: Thực nghiệm và đánh giá kết quả của mô hình IDS cho môi trường IoT

CHƯƠNG 1: TỔNG QUAN VỀ HỆ THỐNG IDS TRONG IOT

1.1. Tổng quan về IoT

Trong bối cảnh bùng nổ dữ liệu, lĩnh vực IoT đang từng bước thể hiện vị thế trong nhiều lĩnh vực trọng yếu như "nhà thông minh (smart home), thành phố thông minh (smart city)", ứng dụng chăm sóc sức khỏe, ứng dụng web, mạng cảm biến không dây, mạng doanh nghiệp, môi trường đám mây, công nghệ lưới điện thông minh. [1] đã ước tính: "Tính đến năm 2030, lượng thiết bị được kết nối dự kiến đạt 24.1 tỷ thiết bị", trong đó ước tính 5,8 tỷ thiết bị được phân bố cho các ứng dụng công nghiệp [2]. Những con số này làm nổi bật sự đa dạng hóa của hệ sinh thái IoT. Tuy nhiên, chúng cũng đặt vấn đề về mối lo ngại bảo mật.

Một nghiên cứu [3] chỉ ra rằng, "các thiết bị IoT chiếm 30% các cuộc tấn công xảy ra trên mang di dộng và mạng không dây". Nghiên cứu khẳng định " Bảo mật có vai trò quan trọng trong môi trường IoT". Hệ thống phát hiện xâm nhập IDS như giải pháp hiệu quả cho vấn đề này. Nhưng với sự phức tạp của kiến trúc IoT cùng các cảm biến, mô-dun truyền thông phong phú dẫn đến cảnh quan phức tạp và biến đổi liên tục. Điều này gây trở ngại lớn cho vấn đề chống tấn công, lấy cắp thông tin, đặc biệt là tại các biên mạng, thiết bị người dùng. Việc áp dụng AI trong phát hiện, cảnh báo tấn công, đặc biệt trên biên mạng giúp giảm thiểu công việc, tăng cao hiệu suất cũng như giảm chi phí vận hành. Chính vì vậy việc áp dụng, kết hợp công nghệ AI, điện toán biên vào hệ thống IDS là bức thiết cho hệ thống IoT nơi có các điều kiện bảo mật khắt khe và nghiêm ngặt.

1.1.1.Lý thuyết về IoT (Internet Vạn Vật)

Khái niệm của Internet of thing (IoT) được Kevin Ashton định nghĩa lần đầu tiên khi để xuất ứng dụng RFID, nhấn mạnh khả năng "tự động hóa thu thập dữ liệu của máy tính" [4]. IoT được biết đến là "một mạng lưới các thiết bị vật lý kết nối với nhau" [4] thông qua các công nghệ truyền thông, có khả năng thu thập, xử lý trao đổi dữ liệu từ thế giới vật lý, ảo, phản ứng với sự can thiệp tối thiểu từ con người. Hiểu một cách đơn giản, IoT là kết nối "vạn vật" với internet. IoT mở ra cuộc cách mạng

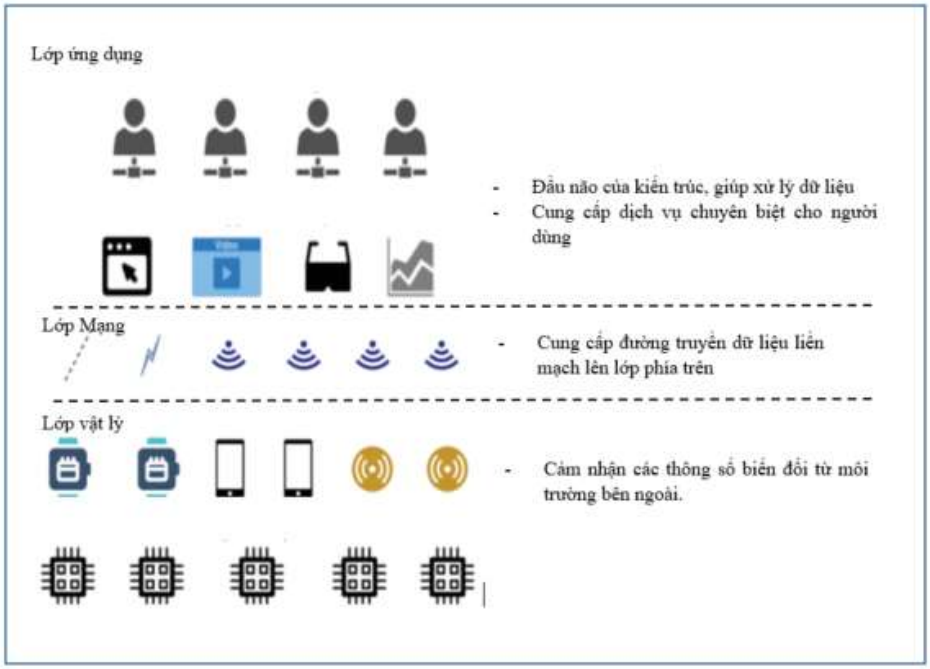
công nghệ toàn cầu, phạm vi kết nối internet được mở rộng. Thay vì giới hạn ở PC, smartphone, laptop, internet kết nối hầu hết các thiết bị, tạo nên mạng lưới khổng lồ.

Sự hình thành và trưởng thành của Internet of Things (IoT) gắn liền với quá trình tiến hóa của công nghệ truyền thông và tính toán. Từ những hệ thống truyền tin sơ khai của thế kỷ XIX đến sự xuất hiện của mạng Internet và World Wide Web vào cuối thế kỷ XX, nền tảng cho một thế giới kết nối tự động đã dần được định hình. Trên cơ sở đó, khái niệm IoT được giới thiệu vào năm 1999 và "chỉ trong vòng hơn một thập kỷ, IoT đã nhanh chóng trở thành một lĩnh vực công nghệ trọng tâm nhờ sự xuất hiện của các nền tảng điện toán nhỏ gọn, chi phí thấp như Raspberry Pi, cùng sự phổ cập mạnh mẽ của thiết bị thông minh vào khoảng năm 2014"[4].

Tuy nhiên, sự phát triển theo chiều rộng với tốc độ cao đã khiến bảo mật trở thành điểm yếu mang tính hệ thống của IoT. Nhiều thiết bị được triển khai với năng lực xử lý hạn chế, thiếu cập nhật phần mềm và không tuân thủ các chuẩn an ninh tối thiểu. Điều này dẫn đến hàng loạt sự cố nghiêm trọng, tiêu biểu là các cuộc tấn công mạng quy mô lớn dựa trên việc chiếm quyền điều khiển thiết bị IoT không được bảo vệ, cho thấy bề mặt tấn công của IoT ngày càng mở rộng và khó kiểm soát. Trước những rủi ro này, các cơ quan quản lý và tổ chức tiêu chuẩn quốc tế bắt đầu đưa ra các biện pháp mang tính định hướng. Bước ngoặt quan trọng là sự ra đời của các nguyên tắc Secure by Design, nhấn mạnh yêu cầu tích hợp cơ chế bảo mật từ giai đoạn đầu của vòng đời phát triển thiết bị, thay vì bổ sung sau khi sản phẩm đã được triển khai. Trong giai đoạn hiện nay, IoT tiếp tục chuyển dịch theo hướng Internet Thế hệ Tiếp theo (Next Generation Internet – NGI), "đề cao tính an toàn, tính minh bạch và khả năng bảo vệ người dùng, đồng thời tận dụng các công nghệ trí tuệ nhân tạo, học máy và thực tại ảo để tối ưu hóa hệ thống" [3]. Từ góc nhìn bảo mật, tiến trình phát triển của IoT được xem như quá trình chuyển đổi từ một hệ sinh thái chủ yếu hướng tới khả năng kết nối sang một cấu trúc công nghệ đòi hỏi an ninh trở thành trụ cột trung tâm. Sự dịch chuyển này được xem như điều kiện tiên quyết để IoT vận hành bền vững trong môi trường phức tạp như công nghiệp, giao thông thông minh và hạ tầng đô thị quy mô lớn.

1.1.2. Kiến trúc IoT

Sự phát triển không ngừng của 5G, công nghiệp 4.0 đã thúc đẩy cải tiến kỹ thuật trong kiến trúc IoT. Chưa có một kiến trúc đồng nhất nào được thống nhất trên toàn cầu cho IoT. Nhiều kiến trúc IoT khác nhau đã được đề xuất bởi các nhà nghiên cứu. Kiến trúc cơ bản nhất trong IoT là kiến trúc ba lớp được đề cập trong nhiều bài nghiên cứu. Đây là kiến trúc được giới thiệu từ giai đoạn đầu của của các nghiên cứu về IoT. Kiến trúc được cấu thành từ ba lớp như hình 1.1.



Hình 1. 1 Kiến trúc hệ thống IoT 3 lớp

"Lớp cảm biến hay còn gọi là lớp vật lý gồm các cảm biến, thiết bị vật lý với chức năng thu thập dữ liệu từ môi trường"[7]. Đây là lớp cảm nhận các thông số vật lý, nhận dạng các đối tượng thông minh trong môi trường, được xem như "tai mắt" của hệ thống IoT. Dữ liệu có thể là thời gian, nhiệt độ, âm thanh, hình ảnh... Các dữ liệu được đóng gói gửi đến lớp trên. Dữ liệu được thu thập từ nhiều thiết bị, nút khác nhau nên rất khó ngăn chặn việc chèn thông tin độc hại, không hợp lệ. Bên cạnh đó, điều kiện môi trưởng khắc nghiệt cũng góp phần ảnh hưởng đến tinh toàn vẹn của dữ liệu. Để giải quyết, cần có phương án xác thực và một số thuật toán áp dụng trong lớp cảm biến.

Lớp mạng chịu trách nhiệm truyền thông cho hệ thống. Dữ liệu được truyền qua kết nối có dây (mạng LAN) hoặc không dây như wifi, bluetooth,..... Với vai trò "người vận chuyển ", lớp mạng kết nối thiết bị với nhau vào mạng lưới truyền thông. Ngoài ra nó còn đảm nhiệm vai trò truyền tải và xử lý dữ liệu nhận được từ cảm biến.

Lớp ứng dụng được xem như đầu não của cả hệ thống. Chúng cung cấp dịch vụ ứng dụng chuyên biệt cho người dùng.

Kiến trúc ba lớp đã thể hiện được ý tưởng của IoT. Tuy nhiên nó chưa đầy đủ để phục vụ cho những nghiên cứu chuyên sâu. Để giải quyết tình trạng này, những cải tiến kỹ thuật đã được đề xuất thông qua các bài nghiên cứu kiến trúc IoT. Dựa trên kiến trúc ba lớp phổ biến, [7] đã đề xuất một kiến trúc IoT mới, cung cấp điện toán biên cơ bản, được gọi là kiến trúc ba lớp mở rộng. "Kiến trúc tuân thủ theo mô hình ba lớp bao gồm: lớp thiết bị đầu cuối IoT, lớp điện toán biên và lớp đám mây. Trong đó lớp biên đóng vai trò trung gian quan trọng giữa thiết bị đầu cuối và đám mây"[7].

"Lớp thiết bị đầu cuối bao gồm các cảm biến bộ truyền động và thiết bị thông minh phân tán trong môi trường. Lớp này thu thập dữ liệu thô từ môi trường như nhiệt độ, độ ẩm, trạng thái thiết bị, hình ảnh thật,".... Sau đó dữ liệu được truyền lên lớp biên thay vì truyền thẳng lên đám mây. Điều này giúp giảm tải truyền dẫn, cả thiện độ trễ cho toàn hệ thống. Thiết bị trong lớp đầu cuối có thể nhận kích hoạt các hoạt động một cách linh động từ các lớp thông mình phía trên. Giai đoạn này, dữ liệu được tạo ra rất lớn và liên tục, đặc biệt trong các ứng dụng giám sát video, robot công nghiệp, thực tế ảo/ tăng cường, cảm biến độ phân giải cao. Điều này khiến nhu cầu xử lý gần nguồn dữ liệu là cần thiết.

Lớp biên hoặc sương mù được xem là điểm mới quan trọng nhất trong kiến trúc IoT hiện đại. Lớp này xuất hiện nhằm giải quyết vấn đề về độ trễ, băng thông và khả năng mở rộng mà các hệ thống IoT truyền thống gặp phải. Các nút tính toán đặt tại biên, như bộ định tuyến có khả năng xử lý, gateway thông minh hoặc máy chủ MEC, cho phép thực hiện phân tích và xử lý dữ liệu ngay tại vị trí gần nơi dữ liệu được tạo ra. Cách tiếp cận này rút ngắn đáng kể thời gian phản hồi của hệ thống, giúp

các ứng dụng thời gian thực như điều khiển robot, giám sát an ninh hay vận hành dây chuyền công nghiệp hoạt động ổn định và hiệu quả hơn. Lớp biên đóng vai trò như nền tảng cho các mô hình trí tuệ nhân tạo phân tán. Các mô hình học máy có kích thước nhỏ hoặc được tối ưu hóa có thể được triển khai trực tiếp tại lớp này để thực hiện các tác vụ như phát hiện bất thường, nhận dạng đối tượng, dự đoán sự kiện hoặc phân loại dữ liệu. Điều này giúp giảm áp lực cho đám mây, đồng thời bảo đảm rằng các hoạt động phân tích quan trọng vẫn diễn ra kể cả khi kết nối mạng không ổn định.

Lớp đám mây trong kiến trúc mở rộng duy trì vai trò trung tâm trong quản lý và xử lý ở quy mô lớn. Không giống như kiến trúc truyền thống, đám mây trong mô hình mới không còn phải xử lý toàn bộ dữ liệu thô mà chỉ tiếp nhận dữ liệu đã được chọn lọc hoặc tổng hợp từ lớp biên. Điều này giúp hệ thống đám mây vận hành tốt hơn, đồng thời có khả năng mở rộng không gian lưu trữ và thực hiện các phân tích sâu với khối lượng dữ liệu lớn. Lớp đám mây vẫn là nơi thực hiện huấn luyện các mô hình quy mô lớn trước khi triển khai chúng tại lớp biên để suy luận nhanh. “Vai trò điều phối tài nguyên và quản lý toàn cục của đám mây vẫn được duy trì, đặc biệt trong các hệ thống IoT quy mô lớn như thành phố thông minh hoặc khu công nghiệp tự động hóa”[7].

Sự kết hợp giữa ba lớp này tạo nên một kiến trúc linh hoạt, thích ứng tốt hơn các yêu cầu của hạ tầng IoT hiện đại. Dữ liệu được xử lý theo hướng phân tầng, trong đó thiết bị tập trung vào việc tạo ra dữ liệu, lớp biên thực hiện xử lý cục bộ và lớp đám mây đảm nhiệm phân tích chuyên sâu. “Kiến trúc này đã chứng minh được hiệu quả trong nhiều hệ thống đòi hỏi khả năng phản hồi tức thời và xử lý dữ liệu phức tạp, đồng thời là cơ sở quan trọng cho sự phát triển của IoT thế hệ tiếp theo”[8].

1.1.3 Đặc trưng dữ liệu IoT

Cùng với sự lan rộng của các ứng dụng IoT tạo ra một lượng dữ liệu lớn, chúng tăng trưởng nhanh chóng trong mạng lưới viễn thông. Lượng dữ liệu này có các đặc trưng điển hình của dữ liệu lớn.

Khối lượng dữ liệu*:* Hệ sinh thái IoT tạo ra lượng dữ liệu cực lớn nhờ sự hiện diện của hàng tỷ thiết bị hoạt động liên tục. Điều này tạo nguồn tài nguyên dồi dào

cho nghiên cứu và phân tích dữ liệu. Tuy nhiên, thách thức nằm ở việc lưu trữ, phân phối và huấn luyện mô hình trên các tập dữ liệu có quy mô quá lớn, vượt khả năng xử lý của các hệ thống truyền thống. Điều này thúc đẩy phát triển các kỹ thuật học phân tán, nén dữ liệu và giảm chiều.

Tốc độ sinh dữ liệu*:* Nhiều thiết bị IoT tạo dữ liệu theo thời gian thực hoặc với tần suất cực cao. Trong bối cảnh học máy, tốc độ này yêu cầu các thuật toán không chỉ chính xác mà còn phải có khả năng cập nhật nhanh. Các mô hình học trực tuyến (online learning) và học tăng cường đóng vai trò quan trọng để đảm bảo mô hình thích ứng kịp thời với dòng dữ liệu liên tục.

Tính đa dạng: Dữ liệu IoT được tạo ra từ những thiết bị khác nhau, hoạt động trong các môi trường và mục đích đa dạng. Sự thiếu đồng nhất về giá trị, định dạng và cấu trúc khiến việc trích xuất đặc trưng (feature extraction) và tích hợp dữ liệu trở thành thách thức chính. Trong học máy, điều này đòi hỏi các kỹ thuật tiền xử lý linh hoạt, các mô hình đa phương thức (multimodal learning) hoặc khả năng học các đặc trưng thông qua mạng nơ-ron sâu.

Tính không chắc chắn và nhiễu: Do điều kiện đo lường, giới hạn phần cứng và sự bất ổn của môi trường, dữ liệu IoT thường chứa sai lệch, nhiễu hoặc thậm chí bị thiếu. Với học máy, chất lượng dữ liệu thấp có thể làm giảm hiệu quả mô hình, dẫn đến suy giảm độ chính xác. Vì vậy, các kỹ thuật như phát hiện ngoại lai (outlier detection), xử lý dữ liệu thiếu, chuẩn hóa và làm sạch trở thành bước quan trọng trong chuỗi xử lý.

*Giá trị khai thác:*Mặc dù dữ liệu IoT phức tạp, khả năng trích xuất tri thức của học máy giúp chuyển dữ liệu thô thành thông tin có thể sử dụng, hỗ trợ dự đoán, tối ưu hóa và tự động hóa. Giá trị này được khuếch đại trong các ứng dụng như điều khiển thông minh, phân tích hành vi, dự báo bảo trì và phát hiện bất thường. Tuy nhiên, việc biến dữ liệu thành giá trị đòi hỏi mô hình phải phù hợp với bối cảnh ứng dụng và có khả năng làm việc với dữ liệu không hoàn hảo

Tính biến thiên: Tần suất sinh dữ liệu và cấu trúc dữ liệu IoT có thể thay đổi bất ngờ theo sự kiện hoặc theo phiên bản thiết bị. Với học máy, điều này khiến mô

hình dễ gặp "drift" hiện tượng phân phối dữ liệu thay đổi theo thời gian, làm giảm hiệu quả dự đoán. Do đó, các phương pháp học thích ứng, cập nhật mô hình liên tục và giám sát chất lượng dữ liệu là nhu cầu thiết yếu để duy trì hiệu suất.

1.1.4. Một số công nghệ điển hình trong IoT

Hệ sinh thái IoT được xây dựng dựa trên nhiều công nghệ khác nhau, bao gồm các thiết bị cảm biến và hệ thống nhúng dùng để thu thập dữ liệu, các giao thức truyền thông như Wi-Fi, Bluetooth, ZigBee, LoRaWAN và các giao thức mạng như MQTT, CoAP, TCP/IP nhằm đảm bảo khả năng kết nối và truyền tải dữ liệu. Bên cạnh đó, các nền tảng điện toán đám mây và phân tích dữ liệu hỗ trợ lưu trữ, xử lý và khai thác thông tin phục vụ các ứng dụng thông minh. Trong khuôn khổ đề án, tập trung nghiên cứu các công nghệ học máy và điên toán biên.

a, Điện toán biên

Mục tiêu cốt lõi của điện toán biên là tối ưu hóa hoạt động của mạng IoT bằng cách hạn chế nhu cầu truyền dữ liệu lên đám mây và giảm độ trễ trong quá trình xử lý. Học máy đóng vai trò trung tâm trong mô hình IoT ba lớp mở rộng nhờ khả năng triển khai trực tiếp trên các thiết bị biên, từ đó hỗ trợ hiệu quả các ứng dụng AI yêu cầu phản hồi thời gian thực và độ trễ cực thấp. Điện toán biên bao gồm ba kiến trúc nền tảng: điện toán sương mù (fog computing), điện toán biên di động (MEC) và đám mây nhỏ (micro cloud). Điện toán sương mù được xem như lớp mở rộng của mô hình đám mây truyền thống. Các nút sương mù được bố trí giữa những thiết bị đầu cuối và trung tâm dữ liệu, đảm nhiệm việc thu thập và hợp nhất dữ liệu dị thể đến từ nhiều nguồn. Những nút này đồng thời cung cấp giao diện thống nhất giúp đồng nhất sự không đồng nhất của dữ liệu, tạo thuận lợi cho ứng dụng và người dùng cuối trong quá trình truy xuất. Trong khi đó, MEC được phát triển chuyên biệt cho hệ thống mạng di động. Khác với kiến trúc sương mù, nút MEC được trang bị khả năng tính toán và lưu trữ mạnh hơn, đồng thời tích hợp các cơ chế ảo hóa giúp tùy biến và điều chỉnh chức năng linh hoạt theo yêu cầu của hệ thống. Nhờ đó, MEC cho phép cung cấp các dịch vụ có độ trễ thấp, độ tin cậy cao và khả năng hồi đáp thời gian thực cho thiết bị di động. Kiến trúc thứ ba, đám mây nhỏ, dựa trên những máy chủ ảo hóa đặt

gần người dùng, thường chỉ cách một bước nhảy mạng. Các đám mây nhỏ này có thể lưu trữ tài nguyên tạm thời và thực thi các tác vụ cục bộ, từ đó mang lại khả năng xử lý và thích ứng linh hoạt tương tự MEC. Song song với ba kiến trúc chủ đạo này, nhiều nghiên cứu gần đây tiếp tục mở rộng mô hình điện toán biên nhằm nâng cao hiệu quả năng lượng và cải thiện độ tin cậy của dữ liệu trong hệ sinh thái IoT. Điện toán biên được triển khai thông qua các nút hoặc máy chủ đặt gần người dùng cuối, với mục tiêu chung là giảm độ trễ, tăng năng lực xử lý cục bộ và hỗ trợ dịch vụ theo thời gian thực. Trong kỷ nguyên 5G, những đặc trưng này trở thành nền móng cho các dịch vụ thông minh, vốn phụ thuộc vào khả năng truyền tải và xử lý dữ liệu lớn.

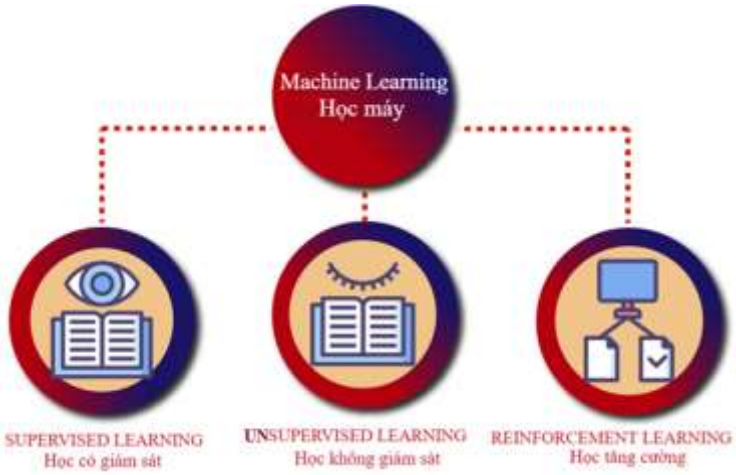
b, Học máy

Internet vạn vật (IoT) kết hợp với học máy (ML) đang tạo nên một hướng tiếp cận mới trong đổi mới dữ liệu, nơi các đối tượng vật lý và không gian số được liên kết thông qua các cảm biến và nền tảng mở rộng hiện đại. Sự hội tụ này đang hình thành một cấu trúc công nghệ toàn cầu mạnh mẽ, hỗ trợ khả năng tương tác, thu thập dữ liệu và phản hồi thông minh trên quy mô lớn. Nhờ đó, IoT đã thay đổi căn bản cách các thiết bị giao tiếp, mở ra các ứng dụng rộng rãi từ nhà thông minh, y tế số cho đến tự động hóa. Tuy nhiên, sự phát triển nhanh chóng của internet vạn vật cũng kéo theo thách thức về khả năng mở rộng, độ tin cậy, mức độ tiêu thụ năng lượng. Để giải quyết các hạn chế này, nhiều nghiên cứu đã khai thác tiềm năng của học máy như một giải pháp cốt lõi giúp tối ưu hóa hoạt động của IoT. ML cho phép hệ thống phân tích dữ liệu phức tạp, nâng cao chất lượng ra quyết định, tự động hóa quá trình vận hành và tối ưu hóa hiệu suất tổng.

Việc ứng dụng ML trong IoT cũng cho phép triển khai các khối chức năng quan trọng như phân tích dữ liệu theo thời gian thực, phát hiện bất thường, ra quyết định tự động, và bảo trì dự đoán. Bằng cách khai thác các mô hình học máy phù hợp, hệ thống IoT có thể tối ưu hóa phân bổ tài nguyên mạng, cải thiện chất lượng dịch vụ và tăng cường khả năng phòng vệ trước các rủi ro an ninh mạng. Sự kết hợp của ML với điện toán biên và điện toán đám mây tiếp tục thúc đẩy năng lực xử lý, giúp IoT đạt hiệu quả cao hơn cả về tốc độ lẫn khả năng mở rộng. Học máy nổi lên như một

phần tử quan trọng để nâng cao hiệu quả và độ tin cậy của hệ sinh thái IoT. Điều này mở ra triển vọng cho các giải pháp IoT thông minh hơn, an toàn hơn và mang tính tự động hóa cao hơn trong tương lai gần.

Các nghiên cứu cho thấy những kỹ thuật học máy khác nhau mang lại hiệu quả riêng khi được triển khai trong môi trường IoT đa dạng về cấu trúc dữ liệu và điều kiện vận hành. Một số nhóm phương pháp tiêu biểu có phạm vi ứng dụng rộng như học có giám sát, học không giám sát, học tăng cường.



Hình 1. 2 Phân loại mô hình học máy

"Học máy có giám sát được biết đến là một trong những kỹ thuật ML phổ biến nhất trong ứng dụng IoT, các mô hình được đào tạo trên tập dữ liệu được gắn nhãn để phân loại hoặc dự đoán" [8]. Học máy giám sát có thể sử dụng để phát hiện lưu lượng bất thường trong mạng IoT, dự báo mẫu bất thường trên các cảm biến, xác định lỗi tiềm ẩn khi chúng xảy ra. Hoặc trong nhà thông minh, hệ thống cũng giúp dự báo năng lượng tiêu thụ dựa trên dữ liệu lịch sử, giúp dễ dàng phân bổ tài nguyên và tiết kiệm năng lượng. Khi dữ liệu đầu vào mang tính đều đặn và không bị biến động mạnh, học có giám sát cho độ chính xác cao, ổn định và chi phí tính toán thấp. Tuy nhiên, hiệu suất của mô hình dễ suy giảm khi môi trường hoạt động thay đổi hoặc khi tập dữ liệu gán nhãn không phản ánh đầy đủ những tình huống hiếm gặp. Điều này khiến các mô hình cần được cập nhật hoặc huấn luyện lại định kỳ để duy trì hiệu năng.

Khác với học có giám sát, các kỹ thuật học không giám sát không chịu phụ thuộc vào dữ liệu được gán nhãn mà dựa vào khả năng tự phát hiện cấu trúc và mẫu ẩn trong tập dữ liệu. Các thuật toán như K-means, phân cụm phân cấp thường được sử dụng để nhóm các thiết bị có hành vi tương tự hoặc nhận diện những tín hiệu bất thường mà không cần biết trước loại sự kiện. Ngoài ra, các phương pháp giảm chiều như phân tích thành phần chính (PCA) được dùng để nén dữ liệu cảm biến có kích thước lớn, giúp tăng tốc độ xử lý và tiết kiệm băng thông truyền tải. Ưu thế lớn nhất của học không giám sát nằm ở khả năng thích nghi với những hệ thống IoT quy mô lớn, nơi lượng dữ liệu tạo ra vượt xa khả năng gán nhãn thủ công. Mặc dù vậy, việc đánh giá hiệu quả thường mang tính tương đối vì không có nhãn chuẩn để so sánh, và chất lượng kết quả phụ thuộc vào lựa chọn tham số cũng như đặc điểm phân bố của dữ liệu.

Bên cạnh đó, học tăng cường nổi lên như một hướng tiếp cận hữu hiệu cho các tình huống đòi hỏi tối ưu hóa hành động theo thời gian thực. Các thuật toán như Q-learning hoặc Deep Q-Network cho phép mô hình tự học thông qua quá trình tương tác với môi trường, từ đó dần tìm được chiến lược tối ưu. Trong lĩnh vực IoT, học tăng cường được ứng dụng để phân bổ tài nguyên mạng, điều chỉnh công suất truyền hoặc xác định chu kỳ hoạt động của cảm biến sao cho tiết kiệm năng lượng nhất. Điểm mạnh của phương pháp này là khả năng thích ứng với môi trường động, vốn là đặc trưng của nhiều mạng IoT không dây. Tuy nhiên, quá trình huấn luyện đòi hỏi thời gian dài và đôi khi cần sức mạnh tính toán đáng kể, đặc biệt khi sử dụng mô hình sâu, khiến việc triển khai trực tiếp lên thiết bị biên gặp nhiều khó khăn.

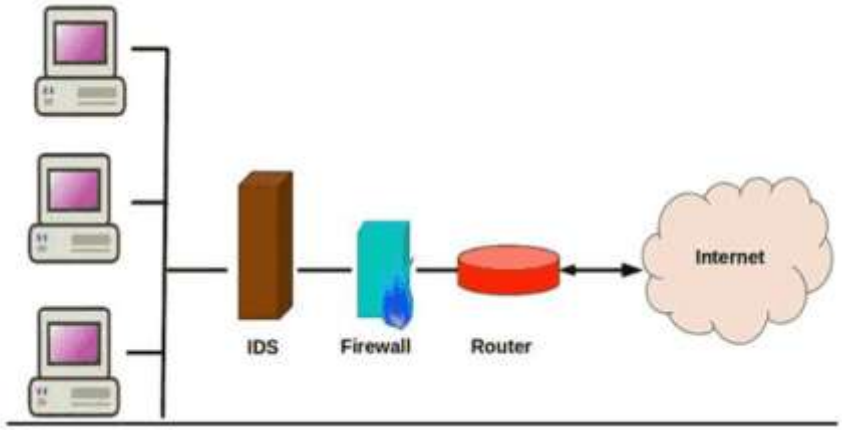
1.2. Tổng quan hệ thống IDS

1.2.1. Lý thuyết về Hệ thống Phát hiện Xâm nhập (IDS)

a, Hệ thống phát hiện xâm nhập IDS

Hệ thống phát hiện xâm nhập được phát triển như một lớp phòng thủ chuyên dụng nhằm nhận diện các hành vi độc hại bên trong hoặc từ bên ngoài mạng máy tính. Khác với tường lửa vốn chủ yếu kiểm soát truy cập dựa trên những quy tắc cố định, IDS thực hiện quá trình phân tích sâu vào hành vi mạng và hoạt động của người

dùng, từ đó phát hiện những bất thường mà các cơ chế bảo mật truyền thống khó nhận biết. IDS có thể tồn tại dưới dạng phần cứng, phần mềm hoặc kết hợp cả hai, vận hành như một "hệ thần kinh" giúp hệ thống kịp thời nắm bắt các dấu hiệu xâm nhập từ đó đưa ra cảnh báo hoặc kích hoạt phản ứng bảo vệ.



Hình 1. 3. Thành phần hệ thống IDS

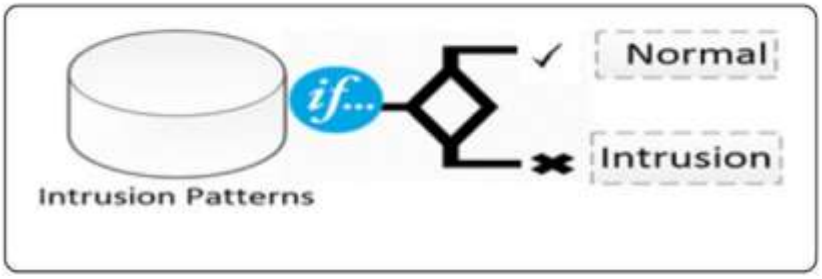
Mục tiêu trọng tâm của IDS là xác định các mẫu lưu lượng và hành vi sử dụng có thể dẫn đến nguy cơ tấn công, chẳng hạn như truy cập trái phép, quét cổng, dò mật khẩu, khai thác lỗ hổng hay các hoạt động phá hoại dịch vụ. Nhờ khả năng quan sát toàn diện đối với lưu lượng mạng và nhật ký hệ thống, IDS đóng vai trò bổ trợ quan trọng cho các giải pháp bảo mật khác, đảm bảo rằng những hành động có khả năng đe dọa tính sẵn sàng, tính toàn vẹn hoặc tính bí mật vẫn có thể được phát hiện cả khi chúng vượt qua các vòng kiểm soát truyền thống.

b, Phân loại hệ thống IDS

- Hệ thống phát hiện xâm nhập dựa trên chữ ký (SIDS)

"Hệ thống phát hiện xâm nhập dựa trên chữ ký dựa trên kỹ thuật so khớp để tìm ra các cuộc tấn công đã biết" [9]. Nó còn được gọi là phát hiện lạm dụng. Với hệ thống này, các phương pháp so khớp được sử dụng để tìm ra sự xâm nhập trước đó, khi chữ ký xâm nhập trùng khớp với chữ ký của cuộc xâm nhập trước đó đã được lưu trong cơ sở dữ liệu, tín hiệu báo động sẽ được kích hoạt ngay lập tức. Hình 1.5 mình họa cách thức họa động của SIDS. Ý tưởng chính của SIDS là xây dựng một cơ sở dữ liệu về các thành phần xâm nhập và so sánh tập hợp các hoạt động chống chống

lại các chữ ký hiện có và từ đó đưa ra cảnh báo khi có sự xâm nhập. SIDS mang lại hiệu quả tốt cho các cuộc tấn công đã tồn tại trong dự liệu.



Hình 1. 4 Cách thức hoạt độn của SIDS

Tuy nhiên với các cuộc tấn công zero-day, SIDS gặp khó khăn lớn do không có chữ ký của các cuộc tấn công này trong cơ sở dữ liệu hiện có. SIDS được sử dụng phổ biến trên các công cụ như Snort, Netstat. SIDS có một điểm yếu cốt lõi chỉ có thể nhận diện những tấn công có mẫu trùng khớp hoàn toàn hoặc gần hoàn toàn với các chữ ký có sẵn. Khi kẻ tấn công tạo ra một biến thể mới bằng cách thay đổi nhẹ cấu trúc hoặc hành vi của mã độc, hệ thống thường không thể phát hiện biến thể này mặc dù nó vẫn duy trì mục đích và cơ chế tấn công tương tự. Điều này khiến SIDS đặc biệt dễ bị qua mặt bởi các loại malware đa hình hoặc các biến thể zero-day chưa từng được ghi nhận. Trong các hệ thống IoT nơi nhiều thiết bị sử dụng firmware lỗi thời, thiếu cơ chế cập nhật việc SIDS không thể nhận diện các hành vi chưa có mẫu càng làm gia tăng nguy cơ xâm nhập. Một hạn chế khác là SIDS không thể phát hiện các cuộc tấn công đa bước hoặc các chiến dịch tấn công kéo dài theo thời gian. Do chỉ tập trung vào các sự kiện đơn lẻ, hệ thống không phân tích được ngữ cảnh, mối liên hệ hay tiến trình hành vi tấn công, dẫn đến việc bỏ sót những chuỗi hoạt động nguy hiểm nhưng không có dấu hiệu rõ ràng trong từng giai đoạn riêng biệt. SIDS thường chỉ đưa ra cảnh báo khi phát hiện mẫu trùng khớp mà không cung cấp thông tin chiều sâu liên quan đến bản chất, động cơ hoặc cơ chế vận hành của cuộc tấn công. Điều này hạn chế khả năng phân tích chiến thuật và đánh giá rủi ro của nhà quản trị hệ thống, đồng thời khiến các quy trình ứng phó sự cố trở nên bị động. Sự gia tăng liên tục của số lượng chữ ký trong cơ sở dữ liệu dẫn đến việc tiêu thụ nhiều tài nguyên tính toán hơn, làm giảm hiệu năng hệ thống và tăng thời gian xử lý. Đối với môi

trường IoT bị giới hạn khả năng tính toán, bộ nhớ và năng lượng, việc triển khai SIDS quy mô lớn đôi khi trở nên không khả thi hoặc làm suy giảm đáng kể hiệu suất của thiết bị.

- Hệ thống phát hiện xâm nhập dựa trên bất thường (AIDS)

Hệ thống phát hiện xâm nhập dựa trên bất thường đang trở thành trọng tâm nghiên cứu, chủ yếu nhờ khả năng khắc phục những giới hạn cố hữu của phương pháp dựa trên chữ ký. Trong cách tiếp cận này, hệ thống xây dựng một mô hình đại diện cho trạng thái bình thường của thiết bị hoặc mạng, thông qua các kỹ thuật như học máy, phân tích thống kê hay mô hình hóa tri thức. Khi hệ thống vận hành, mọi hành vi quan sát được sẽ được so sánh với mô hình hành vi chuẩn; những sai lệch đáng kể so với mẫu được xem là bất thường và có khả năng là dấu hiệu của hành vi xâm nhập. Cơ sở lý thuyết của AIDS dựa trên giả định rằng hành vi độc hại luôn tồn tại những khác biệt so với hành vi hợp lệ của người dùng hoặc thiết bị.

Quy trình phát triển một hệ thống AIDS thường bao gồm hai giai đoạn. Trong giai đoạn huấn luyện, luồng dữ liệu bình thường được thu thập để xây dựng mô hình mô tả hành vi chuẩn. Sang giai đoạn kiểm thử, dữ liệu mới được đưa vào nhằm đánh giá khả năng tổng quát hóa của mô hình đối với các loại xâm nhập chưa từng xuất hiện trong tập huấn luyện. Tùy theo phương pháp học được áp dụng, AIDS được phân loại thành nhiều nhánh, bao gồm các kỹ thuật thống kê hoặc học máy. Mỗi nhóm mô hình đều mang những đặc tính riêng, thích hợp với từng môi trường triển khai và loại dữ liệu khác nhau. Ưu điểm quan trọng nhất của AIDS là khả năng nhận diện các cuộc tấn công zero-day những dạng tấn công chưa từng được ghi nhận trước đó. Vì hệ thống không dựa vào cơ sở dữ liệu chữ ký, nó có thể phát hiện các sự kiện bất thường chỉ dựa trên mức độ sai lệch so với hành vi thông thường. Ngoài ra, AIDS đặc biệt hiệu quả trong việc giám sát và phát hiện các hành vi nguy hiểm xuất phát từ bên trong hệ thống. Khi một tài khoản bị chiếm quyền và thực hiện những thao tác không phù hợp với hồ sơ hành vi gốc, hệ thống có thể đưa ra cảnh báo kịp thời. Việc mô hình hóa hành vi dựa trên hồ sơ tùy biến cũng khiến kẻ tấn công khó đoán được ngưỡng "bình thường", từ đó hạn chế khả năng né tránh hệ thống giám sát.

Tuy vậy, AIDS vẫn tồn tại hạn chế nhất định, đặc biệt là tỷ lệ báo động giả cao. Một số hành vi mới xuất hiện trong hệ thống dù hợp lệ vẫn có thể bị xem là bất thường, khiến việc vận hành yêu cầu cơ chế tinh chỉnh liên tục. Sự thiếu thống nhất trong phân loại phương pháp phát hiện bất thường cũng dẫn đến khó khăn trong đánh giá và so sánh các mô hình. Trong bối cảnh đó, nhiều nghiên cứu đề xuất các nhóm kỹ thuật dựa trên đặc trưng, bao gồm: phương pháp thống kê, dựa trên mẫu, dựa trên luật, dựa trên trạng thái và phương pháp heuristic. Mỗi nhóm mô hình đại diện cho một quan điểm tiếp cận khác nhau trong việc mô tả và nhận diện hành vi bất thường, tạo nên hệ sinh thái đa dạng các kỹ thuật phát hiện xâm nhập phục vụ nhiều nhu cầu thực tiễn.

1.2.2. Hệ thống IDS dựa trên học máy tại biên

Hệ thống phát hiện xâm nhập (IDS) triển khai tại biên mạng ngày càng đóng vai trò quan trọng trong việc bảo vệ các hệ thống IoT trước sự gia tăng về số lượng và mức độ tinh vi của các cuộc tấn công mạng. Không giống các mô hình IDS truyền thống phụ thuộc vào xử lý tập trung trên đám mây, IDS tại biên tận dụng khả năng tính toán cục bộ của các thiết bị biên hoặc gateway để phát hiện tấn công thời gian thực, từ đó giảm đáng kể độ trễ xử lý, tiết kiệm băng thông truyền dữ liệu và tăng cường khả năng phản hồi nhanh trong môi trường phân tán. Song song với đó, tích hợp học máy vào biên cho phép triển khai và thực thi các mô hình học máy ngay tại nơi dữ liệu được tạo ra, như cảm biến IoT, điện thoại thông minh, các thiết bị nhúng. Cách tiếp cận này giúp khắc phục hạn chế về độ trễ, băng thông và quyền riêng tư khi phải truyền toàn bộ dữ liệu lên cloud. Nhờ vậy, Edge ML trở thành nền tảng lý tưởng để triển khai IDS tại biên. Có thể nói IDS tại biên mạng kết hợp với học máy mang lại nhiều lợi thế vượt trội: khả năng phát hiện tấn công gần như thời gian thực, giảm tải cho đám mây, tối ưu tài nguyên mạng, đồng thời tăng cường bảo mật của dữ liệu. Tuy vậy, việc lựa chọn mô hình ML phù hợp, tối ưu hóa tài nguyên và giảm báo động giả vẫn là những yếu tố then chốt trong việc đảm bảo khả năng ứng dụng thực tế của hệ thống.

1.3. Thách thức của các mô hình IDS trong môi trường IoT

- Hạn chế về tài nguyên tính toán và lưu trữ

Các thiết bị IoT được thiết kế với mục tiêu tối ưu về thời gian, chi phí, kích thước và mức tiêu thụ năng lượng. Chính những ràng buộc này khiến chúng trở nên hạn chế đáng kể về khả năng tính toán và không gian lưu trữ, dẫn đến việc triển khai các cơ chế bảo mật tiên tiến gặp nhiều khó khăn. Đối với phần lớn thiết bị IoT giá rẻ, việc đáp ứng đầy đủ tiêu chuẩn bảo mật truyền thống vốn đòi hỏi tài nguyên cao gần như không khả thi. Điều này mở ra nhiều lỗ hổng cho kẻ tấn công tiếp cận. Việc áp dụng các thuật toán truyền thống vốn yêu cầu năng lực tính toán vượt trội thường được xem là không phù hợp với thiết bị biên bị hạn chế về tài nguyên.

Để khắc phục những hạn chế này là ứng dụng các mô hình học máy kết hợp với kỹ thuật tối ưu hóa siêu tham số hiện đại. Việc xử lý và phát hiện bất thường tại chính thiết bị thay vì trên nền tảng đám mây mang lại độ trễ thấp hơn, giúp hệ thống phản ứng nhanh chóng khi xảy ra hành vi tấn công. Như vậy, khi được thiết kế và tinh chỉnh đúng cách, các kỹ thuật Học sâu hoàn toàn có thể thích ứng với môi trường tài nguyên hạn chế của IoT, đồng thời cung cấp khả năng chống lại các mối đe dọa mạng hiện đại hiệu quả.

- Yêu cầu xử lý dữ liệu thời gian thực

Phân tích và xử lý dữ liệu theo thời gian thực là một yêu cầu tất yếu với các mô hình phát hiện xâm nhập khi được triển khai trong môi trường IoT. Tuy nhiên, đa số các mô hình IDS trong các nghiên cứu hiện nay chủ yếu hoạt động theo chế độ ngoại tuyến. Trong lĩnh vực phân loại phần mềm độc hại, thách thức trọng yếu nằm ở việc nhận diện các mẫu đặc trưng để phân biệt giữa lưu lượng bình thường và các lưu lượng tấn công. Khi được triển khai ở chế độ ngoại tuyến, các mô hình học máy chỉ xử lý các tập dữ liệu tĩnh và không phản ánh được sự biến đổi liên tục của luồng dữ liệu trong môi trường vận hành thực tế. Ngược lại, trong bối cảnh dữ liệu luân chuyển liên tục, các hệ thống trực tuyến đòi hỏi cơ chế Học trực tuyến để xử lý, cập nhật và quyết định thực thi theo thời gian thực. Mặc dù phương pháp này có lợi thế lớn trong việc phát hiện các hành vi tấn công mới phát sinh, song cho đến

nay, việc ứng dụng phân tích thời gian thực vẫn chưa được khai thác nhiều trong lĩnh vực phát hiện phần mềm độc hại và IDS cho IoT. Điều này tạo ra khoảng trống lớn giữa mô hình nghiên cứu và yêu cầu thực tiễn trong các hệ thống bảo mật doanh nghiệp hiện đại.

Học Tăng cường được xem là một hướng tiếp cận hứa hẹn nhằm đáp ứng yêu cầu phân tích theo thời gian thực. Phương pháp này cho phép mô hình có thể cập nhật liên tục dựa trên những đặc trưng mới phát sinh, đồng thời vẫn duy trì được các thông tin quan trọng đã học từ trước. Nhờ đó, hệ thống có thể thích ứng tốt với các xu hướng tấn công mới mà không cần trải qua quá trình huấn luyện lại từ đầu vốn vừa tốn thời gian vừa đòi hỏi nhiều tài nguyên.

1.4. Kết Luận chương

Chương 1 đã trình bày cái nhìn tổng quan về hệ sinh thái IoT, Hệ thống phát hiện xâm nhập IDS, và sự kết hợp của IDS, học máy tại biên. Qua đó thể hiện tính thiết thực của hệ thống IDS khi triển khai tại biên mạng. Tuy nhiên bên cạnh những lợi ích được đề cập như tăng tốc độ xử lý, tối ưu tài nguyên và bảo mật dữ liệu, hệ thống cũng gặp nhiều thách thức khi biên mạng bị hạn chế về tài nguyên rất lớn. Thiết bị biên gần như không thể đáp ứng các mô hình học máy phức điều này gây cản trở trong việc phát triển hệ thống. Việc tích hợp các phương pháp tối ưu hóa mô hình được nhận định là phương pháp hiệu quả cho bài toán hiện tại. Các kỹ thuật giảm chiều hay phân cụm dữ liệu không chỉ giảm thiểu tài nguyên mà còn giải quyết rất tốt bài toán tối ưu, đáp ứng tốt nhu cầu của hệ thống IoT hiện tại. Chương 2 sẽ tập trung thiết kế và đề xuất “Hệ thống phát hiện tấn công trong IoT ứng dụng học máy”

CHƯƠNG 2 - HỆ THỐNG PHÁT HIỆN TẤN CÔNG TRONG IOT DỰA TRÊN HỌC MÁY

2.1. Hệ thống IDS dựa trên học máy cho thiết bị IoT tại biên

2.1.1. Cơ sở khoa học của giải pháp đề xuất

a, Cơ sở khoa học các phương pháp xử lý dữ liệu

Trong xử lý dữ liệu cho hệ thống phát hiện xâm nhập, phân cụm và giảm chiều là hai kỹ thuật quan trọng nhằm nâng cao chất lượng dữ liệu và tăng hiệu suất cho mô hình. Trong đó, "phân cụm giữ vai trò nổi bật vì có khả năng tổ chức lại dữ liệu theo cấu trúc tự nhiên, giúp nhận diện các nhóm hành vi tương đồng và tách biệt rõ giữa lưu lượng bình thường và lưu lượng độc hại"[13]. Phân cụm cũng hỗ trợ xử lý hiệu quả vấn đề mất cân bằng dữ liệu bằng cách xác định các vùng có mật độ tấn công cao và trích xuất các mẫu đại diện phù hợp. Phương pháp này cũng loại bỏ nhiễu và các điểm ngoại lai, từ đó nâng cao tính ổn định của dữ liệu đầu vào. Phân cụm có khả năng giảm không gian tìm kiếm và cho phép huấn luyện dữ liệu theo từng cụm, từ đó góp phần giảm chi phí tính toán so với xử lý toàn bộ dữ liệu gốc. Những nghiên cứu về phương pháp phân cụm.

[15] đã thực hiện thuật toán phân cụm FCM và đạt được kết quả tốt, tăng tốc thời gian xử lý so với các mô hình học máy truyền thống.

[16] đánh giá thuật toán K-mean với các trọng số khác nhau được tính toán một cách linh hoạt. Điều này đã khiến mô hình trở nên chất lượng, phục vụ tốt cho các mô hình học máy.

Nghiên cứu chỉ ra tính hiệu quả của phương pháp phân cụm trong học máy. Với tiêu chí thiết kế hệ thống nhẹ, dễ dàng triển khai tại biên mạng, đề án đề xuất phương pháp phân cụm fuzzy C-mean.

b, Cơ sở khoa học về học máy trong hệ thống IDS

Việc kết hợp các mô hình học máy trong hệ thống phát hiện xâm nhập dựa trên cơ sở khoa học rằng lưu lượng mạng chứa các mẫu hành vi lặp lại và có thể được mô hình hóa thông qua quan hệ thống kê giữa các đặc trưng. Các mô hình học máy có khả năng học những đặc trưng tiềm ẩn từ dữ liệu, từ đó phân biệt hiệu quả giữa

hoạt động bình thường và các hình thức tấn công dựa trên sự khác biệt của phân bố đặc trưng. Các mô hình này có khả năng tổng quát hóa, cho phép nhận diện cả những kiểu tấn công chưa từng xuất hiện trong dữ liệu huấn luyện thông qua việc phát hiện các điểm lệch so với mô hình hành vi hợp lệ. Chúng hỗ trợ rất tốt trong xử lý dữ liệu khối lượng lớn của môi trường IoT và mạng hiện đại bằng cách khai thác cấu trúc dữ liệu đa chiều, đồng thời hạn chế sai sót do việc phân tích thủ công. Việc kết hợp học máy với các kỹ thuật tiền xử lý như phân cụm hoặc trích xuất đặc trưng còn tăng cường khả năng nhận diện các mẫu hiếm, giảm nhiễu và tối ưu hóa thời gian suy luận, qua đó cải độ chính xác và tính hiệu quả của IDS trong môi trường thực tế. Trong môi trường biên cần tối ưu hóa về tài nguyên, thời gian huấn luyện, các mô hình học máy đơn giản đã chứng tỏ lợi thế của chúng.

[12] đã đề xuất phương pháp snote cân bằng dữ liệu, sau đó dùng mô hình random forest phân loại tấn công. Kết quả chứng minh hiệu suất mạnh mẽ của mô hình với độ chính xác cao (accuracy = 91.89%, precision = 92%).

[13] đã đề xuất mô hình Decision tree cho kết quả tối ưu đạt độ chính xác cao (Precision= 099%). Điều này chứng tỏ tính hiệu quả của mô hình này với tập dữ liệu CICIoT2023.

Để đáp ứng điều kiện thiết bị tại biên, thuật toán học máy cần có cấu trúc đơn giản, tiêu thụ ít tài nguyên và thời gian suy luận ngắn nhằm đáp ứng yêu cầu phát hiện theo thời gian thực. Các mô hình học máy dạng cây sở hữu cấu trúc đơn giản, nhẹ và yêu cầu tài nguyên tính toán thấp, do đó đặc biệt phù hợp để triển khai trên các thiết bị. Trên cơ sở những phân tích và nghiên cứu đã trình bày, đề án lựa chọn hai mô hình có độ tin cậy và hiệu quả cao là Random forest và Decision tree để thiết kế hệ thống phát hiện xâm nhập (IDS), nhằm đảm bảo khả năng phát hiện chính xác đồng thời tối ưu cho môi trường IoT.

2.1.2. Hệ thống IDS ứng dụng học máy đề xuất

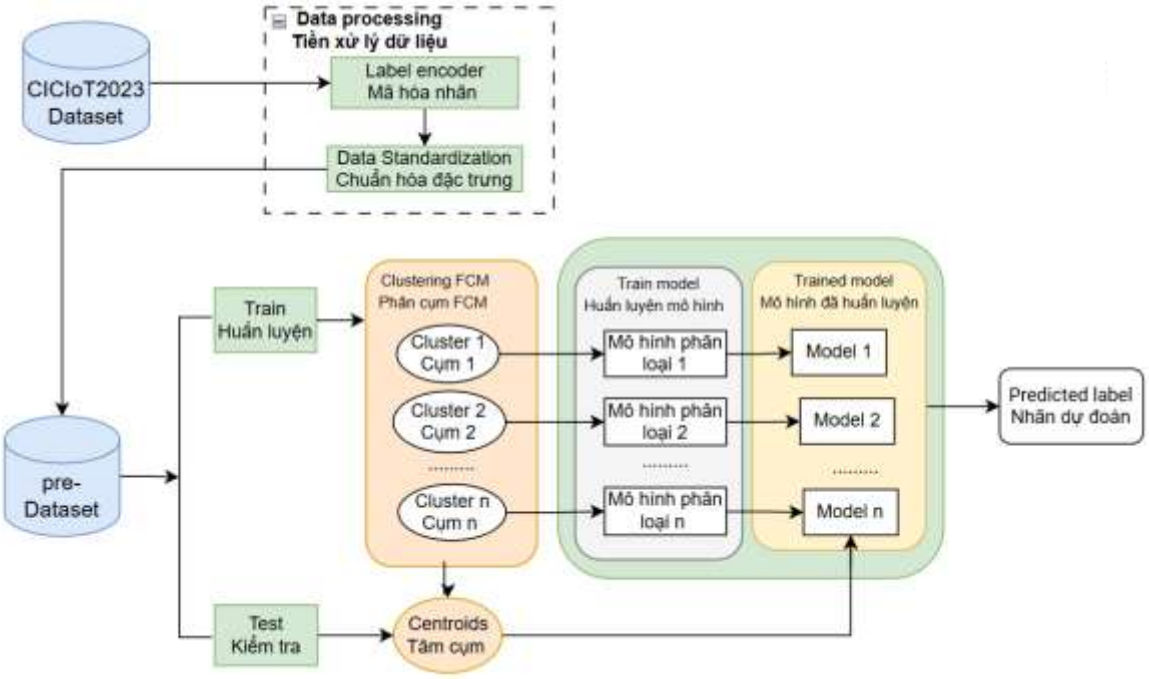
Trên cơ sở khoa học được trình bày ở các mục 2.1.1, đề án đề xuất hệ thống IDS hướng tới mục tiêu tối ưu hiệu quả xử lý, đồng thời đáp ứng yêu cầu triển khai trên các thiết bị biên với tài nguyên hạn chế. Hệ thống sử dụng mô hình hỗn hợp đa

chuyên gia. Thay vì đưa dữ liệu trực tiếp vào mô hình huấn luyện, hệ thống đề xuất sử dụng thuật toán phân cụm Fuzzy c-mean như một bộ định tuyến chia nhỏ dữ liệu thành các sub-data với dữ liệu có tính tương đồng nhất định. Điều này giảm nhiễu đáng kể cho các tập dữ liệu nhỏ do đã được phân cụm, có tính tương đồng cao, phân phối dữ liệu tốt hơn. Bước xử lý này giảm gánh nặng tính toán cho các thuật toán học máy tại biên, hoàn toàn có thể áp dụng các thuật toán đơn giản giúp giảm chi phí tính toán, nâng cao hiệu quả cho mô hình.

Kiến trúc của hệ thống bao gồm ba khối chức năng chính: tiền xử lý dữ liệu, phân cụm dữ liệu và phân loại dữ liệu. Hình 2.1 minh họa cấu trúc tổng thể của hệ thống. Cấu trúc phân tầng này đảm bảo mức độ mô-đun hóa cao, cho phép tối ưu hóa từng thành phần một cách độc lập.

Dữ liệu thô được thu thập từ môi trường IoT, xử lý tuần tự qua từng khối chức năng. Trong khối tiền xử lý dữ liệu, hệ thống thực hiện chuẩn hóa các đặc trưng nhằm đảm bảo tính đồng nhất của tập dữ liệu. Quá trình này bao gồm hai bước chính: (1) Mã hóa các thuộc tính dạng phân loại bằng kỹ thuật label encoding để chuyển đổi chúng sang dạng số nguyên, và (2) chuẩn hóa các đặc trưng số bằng StandardScaler nhằm đưa giá trị về cùng một miền, giúp tăng tính ổn định cho quá trình huấn luyện. Đối với các thuộc tính dạng phân loại này, nghiên cứu sử dụng label encoding để chuyển đổi dữ liệu sang dạng số nhằm đảm bảo khả năng xử lý của các thuật toán học máy. Phương pháp này không làm gia tăng số chiều đặc trưng, giúp giảm chi phí tính toán và bộ nhớ, đặc biệt phù hợp với hệ thống IDS triển khai tại thiết bị biên có tài nguyên hạn chế. Bên cạnh đó, các mô hình học máy dạng cây như decision tree và random forest không nhạy cảm với mối quan hệ thứ tự giữa các nhãn, label encoding vẫn đảm bảo hiệu quả phân loại mà không gây ảnh hưởng tiêu cực đến kết quả.

Sau khi hoàn tất tiền xử lý, dữ liệu được chia thành hai tập con riêng biệt, phục vụ cho quá trình huấn luyện: tập huấn luyện và tập kiểm thử. Đây là đầu vào của các quá trình huấn luyện tiếp.



Hình 2. 1 Mô hình hệ thống IDS đề xuất

Trong khối phân cụm dữ liệu, đề án đề xuất thuật toán phân cụm mờ Fuzzy C-Means (FCM). Việc sử dụng FCM cho phép xác định mức độ phụ thuộc của từng mẫu vào các cụm khác nhau, từ đó tăng khả năng mô tả và phân biệt các dạng hành vi trong lưu lượng mạng.

Sau quá trình phân cụm, hệ thống tiến hành kiểm tra số lượng mẫu trong từng cụm. Đối với những cụm không đạt ngưỡng dữ liệu tối thiểu cần thiết cho việc huấn luyện, hệ thống sẽ tiến hành ghép cụm với cụm gần nhất và tính toán lại tâm cụm nhằm bảo đảm sự ổn định của quá trình phân tích. Khi dữ liệu đã được tổ chức phù hợp, hệ thống đưa dữ liệu vào khối huấn luyện.

Hiệu năng của hệ thống sau đó được đánh giá toàn diện và so sánh với mô hình không áp dụng phân cụm (tức là ứng dụng mô hình học máy trực tiếp dữ liệu gốc). Hệ thống được đánh giá dựa trên độ chính xác của mô hình thông qua các tham số accury, precision, F1-score.

Quy trình đánh giá được xây dựng nhằm bảo đảm tính khách quan, cung cấp cơ sở khoa học tin cậy để đánh giá hiệu quả của giải pháp IDS dựa trên kết hợp giữa phân cụm dữ liệu và học máy. Các kết quả thu được có vai trò làm nền tảng cho việc

triển khai hệ thống trong thực tế, đặc biệt là trong môi trường IoT có giới hạn về tài nguyên tính toán. Trên cơ sở kiến trúc đã đề xuất, bước tiếp theo tập trung vào việc xây dựng và xử lý bộ dữ liệu huấn luyện có chất lượng cao nhằm nâng cao hiệu quả vận hành của hệ thống IDS.

2.2. Cơ sở lý thuyết của phương pháp phân cụm Fuzzy C-Mean

"Phân cụm được biết đến là thuật toán học máy không giám sát. Mục tiêu chính của phân cụm C-mean là phân vừng dữ liệu thành các cụm tập hợp, mỗi điểm dữ liệu được gán giá trị thành viên" [18]. FCM thường được sử dụng trong nhận dạng mẫu và phân đoạn ảnh. FCM là là phần mở rộng của của thuật toán phân cụm K-mean hay còn gọi là C-mean cứng. Trong thuật toán này, mỗi điểm dữ liệu thuộc về một cụm và quyền thành viên được biểu diễn thông qua một giá trị gọi là mức độ thành viên. "Giá trị này nằm trong khoảng từ 0 đến 1" [18]. Điều này tăng tính linh hoạt cho vệc gán cụm.

- Cơ sở toán học

FCM phân cụm bằng cách tối ưu hóa hàm mục tiêu để tạo ra miền dữ liệu rõ ràng và dày đặc. "Hàm mục tiêu cho bài toán tối ưu được mô tả bằng phương trình sau:

$$J_{f_{cm}}(U,V) = \sum_{i=1}^{c}\sum_{j=1}^{N}(u_{ij})^m D_{\text{ij}}^2, m>1 \qquad (2.1)$$

N: số lượng đối tượng, điểm dữ liệu

c: số lượng phân cụm

$u_{ij:}$ độ phụ thuộc của điểm dữ liệu thứ x_k đối với cụm v_i

D_{ij}^2: khoảng cách từ dữ liệu x_k đến tâm cụm v_i

m: hệ số mờ

Hàm (2.1) tính tổng các khoảng cách giữa tất cả điểm dữ liệu và tâm cụm, từ đó tối ứu hàm mục tiêu để xác định các cụm nhỏ nhất. Ràng buộc chính của hàm mục tiêu là tổng bậc thành viên của một điểm dữ liệu trên các cụm bằng 1:

$$\sum_{i=1}^{c} u_{\text{ij}} = 1 \qquad (2.2)$$

Trong FCM, hệ số mờ (m) quyết định độ mờ của của phân cụm, ảnh hưởng mạnh đến cash dữ liệu được phân bố trong cụm. "Với m =1, FCM tiệm cận k-means, mỗi điểm dữ liệu chỉ thuộc về một cụm duy nhất, ma trận membership trở nên sắc nét"[18]. Khi độ mờ tăng lên, các điểm dữ liệu được phân bố mềm hơn và được phân vào nhiều cụm. Trong hầu hết các thuật toán cũng như FCM, hệ số mờ thường được đặt là 2.

Đại lượng D_{ij}^2 thể hiện bình phương khoảng cách giữa điểm dữ liệu x_j đến vị trí tâm cụm v_i. Khoảng cách này có thể tính bằng nhiều công thức toán học khác nhau. Trong khuôn khổ đề án đề xuất khoảng cách Cosine.

Khoảng cách Cosine dựa trên độ tương đồng vector, giúp đo lường sự khác biệt dựa trên hướng thay đổi vì độ lớn. Khoảng cách này được định nghĩa dựa trên cosine của góc 2 vector x_j và v_i:

$$D_{ij} = 1 - \frac{x_j . v_i}{\| x_j \| . \| v_i \|} \qquad (2.3)$$

Cơ chế tính khoảng cách theo hướng khiến Cosine đặc biệt hữu ích trong các trường hợp dữ liệu có giá trị mang tính tương đối. Ví dụ như chuỗi đặc trưng của lưu lượng mạng, tín hiệu cảm biến hoặc vector biểu diễn văn bản. Trong bối cảnh này, hai điểm dữ liệu có cùng hướng dù khác nhau về độ lớn vẫn được xem là tương tự nhau. Khi kết hợp với FCM, khoảng cách Cosine tạo ra phân cụm theo dạng phân bố góc (angular clustering), giúp thuật toán phân biệt các nhóm dữ liệu dựa trên cấu trúc hình học của hướng thay vì độ lớn.

Khi xác định độ phụ thuộc của điểm dữ liệu, FCM khai thác thông tin giữa các cụm, điều này đảm bảo phân bố mềm của các điểm vào cụm.

$$u_{ij} = \left[\sum_{k=1}^{c} \left(\frac{D_{ij}^2}{D_{kj}^2} \right)^{\frac{1}{m-1}} \right]^{-1} \qquad (2.4)$$

Thuật toán FCM gán cho mỗi điểm dữ liệu x_j một độ phụ thuộc tương ứng u(x,v) với từng tâm cụm v_i. Ở bước khởi tạo, các ràng buộc toán học đặc trưng của mô hình fuzzy clustering được áp dụng nhằm thiết lập tập giá trị độ phụ thuộc ban

đầu để đảm bảo tính hợp lệ và ổn định của quá trình tối ưu. Hoặc sử dụng chiến lược khác là xác định trước vị trí các tâm cụm, mặc dù cách này có thể dẫn đến số vòng lặp lớn hơn trong một số trường hợp, nhưng khi các tâm được lựa chọn phù hợp, tốc độ hội tụ của thuật toán có thể được cải thiện rõ rệt.

Trong mỗi vòng lặp, thuật toán cập nhật độ phụ thuộc của các điểm dữ liệu dựa trên khoảng cách của chúng đến các tâm cụm hiện tại. Hàm mục tiêu được tính toán bằng cách tổng hợp khoảng cách giữa các điểm và các tâm cụm (xét đến trọng số là mức độ phụ thuộc của chúng). Khi giá trị hàm mục tiêu giảm, điều đó phản ánh sự hình thành các cụm ngày càng chặt chẽ và phân biệt rõ ràng hơn. "Các tâm cụm mới được xác định bằng trung bình có trọng số của toàn bộ điểm dữ liệu, trong đó trọng số chính là giá trị độ phụ thuộc sau khi cập nhật" [17]. Sau bước này, tâm cụm được thay đổi tương ứng. Quá trình lặp lại cho đến khi đạt được điều kiện hội tụ.

$$v_i = \frac{\sum_{j=1}^{n} u_{ij}^{m} x_j}{\sum_{j=1}^{n} u_{ij}^{m}} \tag{2.5}$$

[8] đã đề cập đến mã giả của thuật toán FCM. Luồng dữ liệu của thuật toán diễn ra theo cấu trúc lặp, trong đó ma trận độ phụ thuộc và các tâm cụm liên tục được cập nhật đến khi đạt trạng thái hội tụ.

Ở giai đoạn đầu, thuật toán tiếp nhận các tham số đầu vào gồm tập dữ liệu X, số cụm (c), hệ số mờ (m), ngưỡng hội tụ ε, và số vòng lặp tối đa L_{max}. Trên cơ sở đó, ma trận độ phụ thuộc ban đầu (U) được khởi tạo ngẫu nhiên sao cho mỗi điểm dữ liệu có tổng độ phụ thuộc vào tất cả các cụm bằng 1. Ma trận này có vai trò là nguồn thông tin đầu tiên quyết định quá trình hình thành cụm trong các bước tiếp theo.

Ở mỗi vòng lặp, tâm cụm sẽ được cập nhật dựa trên giá trị độ phụ thuộc hiện tại. Đây là bước tổng hợp thông tin từ dữ liệu nhằm hình thành đại diện mới của mỗi cụm tại thời điểm lặp.

Sau khi các tâm cụm được tính toán, thuật toán tiếp tục cập nhật ma trận độ phụ thuộc. Mức độ phụ thuộc của một điểm dữ liệu vào một cụm được xác định từ tỉ lệ khoảng cách của điểm đó đến tâm cụm đang xét so với khoảng cách đến các tâm

cụm còn lại. Nhờ cơ chế chuẩn hóa này, thuật toán khai thác hiệu quả thông tin giữa các cụm, đảm bảo rằng mỗi điểm dữ liệu được phân bố mềm vào tất cả các cụm theo mức độ phù hợp.

Algorithm 1: FCM Algorithm

Input: $X, c, m, \varepsilon, l_{\max}$

$U = random([c, n])$

for $l = 1$ **to** $l_{\max}$ **do**

$$v_i = \frac{\sum_{j=1}^{n} u_{ij}^m x_j}{\sum_{j=1}^{n} u_{ij}^m}$$

$$u_{ij} = \left[\sum_{k=1}^{c} \left(\frac{D_{ij}^2}{D_{kj}^2} \right)^{\frac{1}{m-1}} \right]^{-1}$$

if $\|V^{l-1} - V^l\| \leq \varepsilon$ **then**

Stop

Output: U, V

Hình 2. 2 Luồng dữ liệu hệ thuật toán FCM

Thuật toán đánh giá mức độ thay đổi giữa các tâm cụm ở hai lần lặp liên tiếp thông qua chuẩn hiệu. Nếu sự thay đổi này nhỏ hơn ngưỡng ε nghĩa là thuật toán đã hội tụ và quá trình lặp dừng lại. Ngược lại, chu trình cập nhật độ phụ thuộc và tâm cụm tiếp tục được tiến hành. Kết quả cuối cùng của thuật toán bao gồm ma trận độ phụ thuộc (U) sau hội tụ và tập các tâm cụm (V) tương ứng. Hai thông tin này phản ánh cấu trúc cụm mờ của tập dữ liệu ban đầu và được sử dụng cho các mục đích phân tích hoặc phân loại tiếp theo.

Độ phức tạp thời gian của các thuật toán phụ thuộc vào quy mô dữ liệu. Khi tính toán trong phạm vi vòng lặp và xét đến vòng lặp, độ phức tạp tổng thể của thuật toán được xác định là:

$$O(n) = O((c+d).n.c.l) \quad (2.6)$$

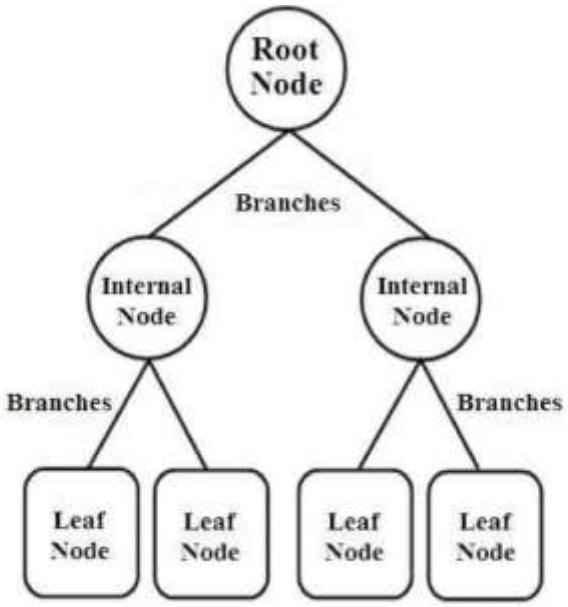
2.3. Cơ sở lý thuyết của mô hình học máy

2.3.1. Mô hình Decision tree

Cây quyết định được biết đến với khả năng diễn giải hiệu quả. Ý tưởng chính của thuật toán là phân vùng dữ liệu bằng phương pháp đệ quy thành các tập con dựa trên giá trị của các thuộc tính khác nhau cho đến khi đáp ứng hàm mục tiêu. Quá trinh này tạo ra cấu trúc dạng cây, mỗi nút đại diện cho một quyết định hoặc phép chia dựa

trên thuộc tính cụ thể. Thuật toán chọn ra thuộc tính tốt nhất để sử dụng cho mỗi phép chia, như tăng mức độ thông tin, tăng tỷ lệ và chỉ số Gini. Cây quyết định có hiệu quả với trong việc xử lý dữ liệu phân loại và số.

Quy trình học của cây quyết định gồm một loạt các bước trong đó dữ liệu được chia thành các tập con đồng nhất như biểu diễn bởi hình 3. Nút gốc đóng vai trò là điểm khởi đầu của cây quyết định và đại diện cho không gian dữ liệu đầu vào. Tại đây, thuật toán lựa chọn đặc trưng cùng giá trị ngưỡng mang lại khả năng phân tách tối ưu theo một tiêu chí đánh giá đã định trước. Việc lựa chọn này được thực hiện nhằm tạo ra khác biệt lớn nhất giữa các nhóm dữ liệu con sau khi chia. Quá trình xây dựng cây diễn ra theo phương thức đệ quy: mỗi tập con được tạo ra từ một lần phân chia lại trở thành một nút trung gian mới và tiếp tục được phân tách dựa trên đặc trưng tối ưu tại nút đó. Chu trình này lặp lại cho đến khi một tiêu chí dừng được thỏa mãn, chẳng hạn như khi nút trở nên thuần nhất hoặc khi độ sâu tối đa của cây đã đạt đến giới hạn cấu hình. Những nút không còn được phân chia thêm nữa được xem như nút lá, và chúng biểu diễn quyết định cuối cùng hoặc nhãn lớp tương ứng. Toàn bộ quá trình lựa chọn đặc trưng và ngưỡng phân chia tại mỗi nút được điều khiển bởi các thước đo định lượng như độ lợi thông tin, chỉ số Gini hoặc mức giảm phương sai, bảo đảm rằng cấu trúc cây được tối ưu hóa theo tiêu chí học tập đã lựa chọn. “Hiệu quả của các phương pháp dựa trên cây quyết định chịu ảnh hưởng đáng kể bởi nhiều yếu tố quan trọng, quyết định khả năng học tập, mức độ diễn giải và tính phù hợp của mô hình đối với các dạng bài toán khác nhau. Các yếu tố cốt lõi này bao gồm chất lượng của dữ liệu đầu vào, độ sâu của cấu trúc cây, tiêu chí được sử dụng để phân tách nút và chiến lược cắt tỉa nhằm kiểm soát mức độ phức tạp của mô hình”.



Hình 2. 3 Cấu trúc mô hình Decision tree

Việc đảm bảo dữ liệu sạch, nhất quán và được xử lý trước bao gồm loại bỏ hoặc hiệu chỉnh giá trị bị thiếu, giảm thiểu ảnh hưởng của ngoại lai có thể nâng cao độ chính xác của mô hình thu được. Thêm vào đó, quá trình lựa chọn và thiết kế đặc trưng đóng vai trò then chốt, bởi các đặc trưng liên quan và được biến đổi hợp lý không chỉ cải thiện chất lượng phân chia tại các nút mà còn góp phần tăng tính tổng quát của mô hình.

a, Quy tắc phân tách

Phương pháp phân tách giúp xác định vị trí cây nên phân tách trong các nút của nó. Từ đó quyết định phương pháp chia phân chia tập dữ liệu dựa trên điều kiện thích hợp. Việc lựa chọn phương pháp phân tách đặc biệt quan trọng, ảnh hưởng đến cấu trúc và hiệu suất của cây quyết định.

Chỉ số Gini

Chỉ số Gini là một tiêu chí phân tách phổ biến trong các thuật toán phân tách của cây quyết định. Gini đo lường xác xuất mẫu được lựa chọn ngẫu nhiên bị phân loại sai nếu được dán nhãn ngẫu nhiên. Chỉ số đánh giá chất lượng mức phân tách và được tính toán cho mỗi phép phân tách tiềm năng trong tập dữ liệu.

$$Gini(S) = 1 - \sum_{i=1}^{n} p_i^2 \qquad (2.7)$$

- S đại diện cho tập dữ liệu
- n: số lớp
- p_i: xác xuất mẫu thuộc lớp i

Giá trị Gini nằm trong khoảng từ 0 (độ tinh khiết hoàn hảo) đến 1 (độ tinh khiết tối đa). Gini có trọng số càng thấp, hiệu suất phân chia càng cao.

Độ lợi thông tin

Độ lợi dựa trên cơ sở entropy trong lý thuyết thông tin. Entropy biểu thị mức độ hỗn loạn hoặc sự khó dự đoán trong phân bố của tập dữ liệu. Khi áp dụng vào quá trình tạo cây, IG đánh giá mức độ hiệu quả của một thuộc tính bằng cách đo lường mức giảm entropy sau khi dữ liệu được phân chia. Nói cách khác, IG tìm kiếm phép tách giúp giảm thiểu sự bất định trong dữ liệu huấn luyện và hướng tới các tập con có tính thuần nhất cao.

$$E(S) = -\sum_{i=1}^{n} p_i \log_2(p_i) \tag{2.8}$$

- n: số lượng lớp phân biệt trong tập dữ liệu
- p_i: tỷ lệ phần tử lớp i trong tập S

Khi đó độ lợi của một phép chia theo thuộc tính A được tính theo công thức:

$$IG(S, A) = E(S) - \sum_{v \in value(A)} \frac{|S_v|}{|S|} E(S_v) \tag{2.9}$$

A đại diện tập các giá trị có thể nhận của thuộc tính A

S_v là tập con của S chứa các mẫu có giá trị thuộc tính bằng v

Biểu thức trên phản ánh mức thay đổi entropy khi chuyển từ tập ban đầu sang các tập con được tạo ra sau khi phân tách theo thuộc tính A. Thuộc tính nào tạo ra độ lợi lớn hơn sẽ được xem là hiệu quả hơn, vì nó mang lại mức độ thuần nhất cao hơn cho các tập con và từ đó cải thiện khả năng phân biệt của cây quyết định.

Độ lợi thông tin

Tỷ lệ độ lợi thông tin (Information Gain Ratio IGR) là phần mở rộng của tiêu chí độ lợi thông tin, nhằm khắc phục xu hướng thiên vị của Information Gain đối với các thuộc tính có số lượng giá trị phân biệt lớn. Bằng cách xét đến lượng nhánh và

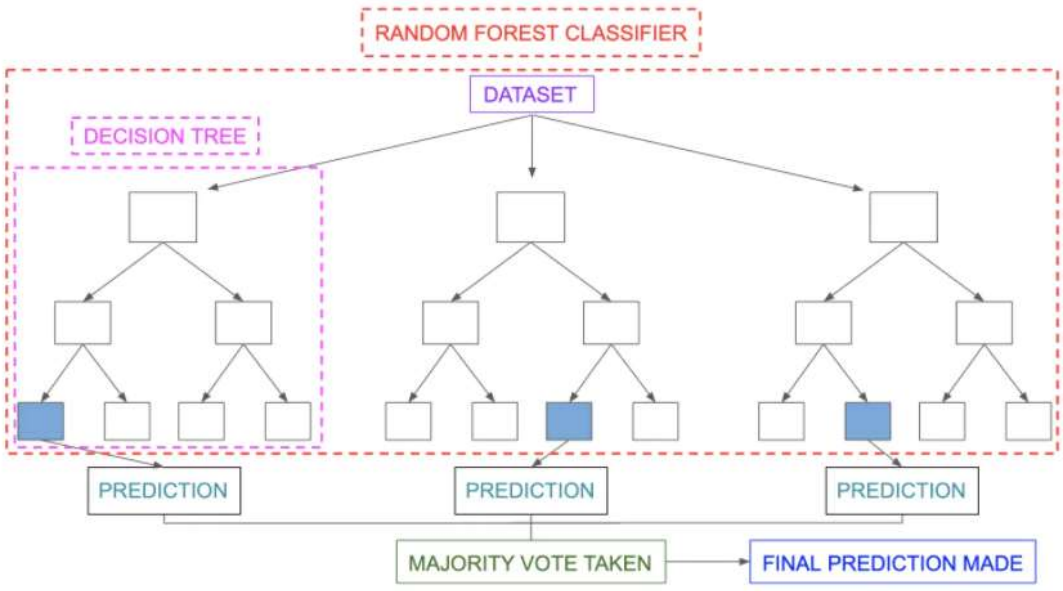
kích thước của chúng trong quá trình phân chia, IGR giảm thiểu khả năng lựa chọn thuộc tính chỉ vì chúng tạo ra nhiều phân vùng nhỏ. Phương pháp này thực hiện bằng cách chuẩn hóa độ lợi thông tin thông qua việc chia cho "thông tin nội tại" hay "thông tin phân chia" (SplitInfo). Việc chuẩn hóa này giúp loại bỏ sự ưu ái đối với các thuộc tính đa giá trị, từ đó tạo ra cấu trúc cây quyết định ổn định và cân bằng hơn. Tỷ lệ độ lợi thông tin cho một phép tách theo thuộc tính A trên tập dữ liệu S được xác định bằng công thức:

$$IGR(S) = \frac{inoformationGain(S, A)}{SplitInfor(S, A)} \tag{2.10}$$

2.3.2. Mô hình Random forest

"Random forest là mô hình học máy có giám sát, xây dựng trên cơ sở các cây quyết định để hình thành mô hình học máy mạnh mẽ hơn. Mô hình lấy kết quả dự đoán của nhiều cây quyết định, dựa trên chúng để đưa ra kết quả dự đoán cuối cùng" [12].

Quy trình xây dựng mô hình được mô tả trong hình 2.5



Hình 2. 4 Cấu trúc mô hình Random forest

(1) Khởi tạo tập cây thông qua lấy mẫu bootstrap

Đầu tiên, thuật toán tiến hành tạo nhiều tập dữ liệu con bằng cách áp dụng kỹ thuật lấy mẫu ngẫu nhiên có hoàn lại từ bộ dữ liệu gốc. Mỗi tập con được hình thành

độc lập và có thể chứa lặp lại một số mẫu.Cơ chế lấy mẫu bootstrap giúp đảm bảo rằng mỗi cây được huấn luyện trên một quan sát khác nhau của dữ liệu, qua đó tăng tính đa dạng giữa các cây thành phần.

(2) Lựa chọn ngẫu nhiên tập đặc trưng tại mỗi điểm phân chia

Trong quá trình hình thành từng cây, Random forest không xét toàn bộ đặc trưng để tìm điểm phân chia tối ưu. Thay vào đó, tại mỗi nút, thuật toán chỉ chọn một tập con ngẫu nhiên và tìm điểm phân tách tốt nhất trong tập con đó. Việc giới hạn không gian đặc trưng theo cách ngẫu nhiên này làm giảm mức độ tương quan giữa các cây, đồng thời hạn chế nguy cơ mô hình học quá sâu vào bất kỳ đặc trưng nổi bật nào trong dữ liệu.

(3) Xây dựng và huấn luyện các cây quyết định độc lập

Mỗi cây quyết định được phát triển trên tập dữ liệu bootstrap và tập đặc trưng. Các cây này được xây dựng độc lập và thường không áp dụng cắt tỉa (pruning), qua đó cho phép chúng mô hình hóa dữ liệu theo những hướng khác nhau. Chính sự đa dạng trong cấu trúc và quá trình học của từng cây là yếu tố giúp Random forest đạt hiệu quả cao trong tổng hợp dự đoán.

(4) Tổng hợp dự đoán của mô hình

Khi nhận dữ liệu mới, mỗi cây trong rừng tạo ra một dự đoán riêng biệt. Random forest sau đó kết hợp các dự đoán này bằng cách:

- Với bài toán phân loại, thuật toán biểu quyết đa số, chọn lớp được dự đoán nhiều nhất.
- Với bài toán hồi qua, tính trung bình các dự báo từ cây.

Cơ chế tổng hợp này giúp giảm phương sai mô hình và cải thiện độ ổn định so với việc sử dụng một cây quyết định đơn lẻ.

(5) Đánh giá và tinh chỉnh mô hình

Mô hình sau khi xây dựng được đánh giá bằng các chỉ số hiệu suất theo từng loại bài toán. Dựa trên kết quả đánh giá, các tham số như số cây, số đặc trưng được lựa chọn tại mỗi nút hoặc độ sâu tối đa của cây có thể được điều chỉnh nhằm tối ưu hiệu quả dự đoán. “Với cơ chế tổng hợp dự đoán từ nhiều cây quyết định, Random

forest giúp giảm đáng kể sai số của các bộ phân loại đơn lẻ, nhờ đó mang lại độ chính xác cao và khả năng khái quát hóa tốt hơn. Tính linh hoạt của thuật toán cho phép nó áp dụng hiệu quả cho cả hai loại bài toán phân loại và hồi quy, đồng thời xử lý tốt nhiều dạng dữ liệu thường gặp trong môi trường IoT như dữ liệu số, dữ liệu phân loại hoặc dữ liệu cảm biến có nhiễu. Đặc biệt, Random forest có khả năng xử lý tập dữ liệu lớn và phân tán, đặc điểm điển hình của các mạng IoT nơi các thiết bị liên tục tạo ra lượng dữ liệu dày đặc theo thời gian thực". Random forest có khả năng học tốt với dữ liệu thiếu hoặc không hoàn chỉnh, cùng với việc hỗ trợ cơ chế đánh giá mức độ quan trọng của các đặc trưng, từ đó giúp các nhà phân tích nhận diện những yếu tố có ảnh hưởng mạnh đến hành vi hệ thống, hỗ trợ tốt cho việc phát hiện bất thường hoặc tấn công mạng IoT.

2.4. Tập dữ liệu CICIoT2023

2.4.1. Nguồn gốc

a, Nguồn gốc dữ liệu

CICIoT2023 là bộ dữ liệu thực tế, quy mô lớn được xây với mục đích thúc đẩy nghiên cứu về bảo mật IoT, đặc biệt là cho các hệ thống IDS. Được phát triển bởi Viện An ninh mạng Canada, "bộ dữ liệu này ghi lại nhiều kịch bản tấn công khác nhau trong môi trường IoT thực tế, trở thành chuẩn mực để đánh giá các mô hình học máy và học sâu trong phát hiện mối đe dọa IoT"[9]. Hình 2.6 minh họa phòng thí nghiệm tại CIC. Các thiết bị được phân bố khắp phòng thí nghiệm, cấu hình trên mạng cục bộ.



Hình 2.5 Phòng thí nghiệm CIC

b, Đặc trưng dữ liệu

Bộ dữ liệu này được xây dựng trong một mô hình mạng thực tế tại phòng thí nghiệm, sử dụng 105 thiết bị IoT thực tế đa dạng đóng vai trò là nạn nhân hoặc kẻ tấn công. "Điểm nổi bật là việc thu thập dữ liệu từ 33 cuộc tấn công khác nhau được phân loại thành bảy nhóm chính, bao gồm DDoS, DoS, Recon, Web-based, brute force, spoofing và Mirai"[9]. Việc các cuộc tấn công thực hiện bởi thiết bị IoT độc hại khiến bộ dữ liệu trở nên sát với môi trường thực tế hơn so với các nỗ lực trước đây. Các tác giả [9] đã "chuyển đổi dữ liệu mạng thô thành các tệp CSV với các tính năng chi tiết, sau đó đánh giá hiệu suất của nhiều thuật toán học máy và học sâu để phân loại lưu lượng độc hại và lành tính". Tập dữ liệu CICIoT2023 được phát triển đáp ứng nhu cầu về một nguồn dữ liệu hiện đại, phản ánh trung thực đe dọa bảo mật trong môi trường Internet of Things (IoT) thực tế. Trong bối cảnh IoT ngày càng mở rộng với mức độ kết nối dày đặc, việc xây dựng các hệ thống IDS hiệu quả đòi hỏi những bộ dữ liệu có độ phức tạp cao, mô phỏng sát các kịch bản tấn công mà thiết bị IoT phải đối mặt. CICIoT2023 ra đời như một tiêu chuẩn mới, hướng đến việc hỗ trợ nghiên cứu, thử nghiệm các giải pháp bảo mật dựa trên học máy và học sâu. Không giống phần lớn các bộ dữ liệu an ninh mạng vốn sử dụng máy tính truyền thống làm nguồn phát tấn công, CICIoT2023 là một trong số ít bộ dữ liệu mà toàn bộ hành vi tấn công đều bắt nguồn từ chính các thiết bị IoT độc hại. Những thiết bị này – điển hình là Raspberry Pi – được cấu hình như các nút tấn công trong mạng, từ đó tạo ra các kịch bản phản ánh đúng bản chất của các mối đe dọa IoT: nhỏ gọn, phân tán và khó kiểm soát. Đặc điểm này có ý nghĩa quan trọng đối với nghiên cứu an ninh IoT. Bởi vì hành vi lưu lượng của thiết bị IoT khác biệt đáng kể so với máy tính, một tập dữ liệu được sinh ra từ chính các thiết bị IoT sẽ giúp các mô hình học máy học được các mẫu giao tiếp thực tế, từ đó cải thiện đáng kể độ chính xác và độ tin cậy khi triển khai trong các hệ thống thực.

Tập dữ liệu CICIoT2023 được công bố dưới hai dạng chính là pcap và csv, đáp ứng linh hoạt nhu cầu của nhiều nhóm nghiên cứu. Với cấu trúc dữ liệu phong

phú, CICIoT2023 đã được sử dụng rộng rãi để đánh giá hiệu suất của nhiều mô hình học máy và học sâu. Các thí nghiệm cho thấy những thuật toán như Random forest và mạng nơ-ron sâu có khả năng đạt độ chính xác cao vượt trội, đặc biệt trong phân loại nhị phân giữa lưu lượng lành tính và độc hại. CICIoT2023 là một cột mốc đáng chú ý trong lĩnh vực bảo mật IoT về quy mô cũng như mức độ chân thực của môi trường thí nghiệm, các đặc trưng được cung cấp. Với việc mô phỏng các cuộc tấn công do chính thiết bị IoT thực hiện trong một cấu trúc liên kết phức tạp, bộ dữ liệu này đã mở ra hướng tiếp cận mới cho việc phát triển và đánh giá các hệ thống phát hiện xâm nhập. Nhờ đó, CICIoT2023 không chỉ đóng vai trò như một bộ dữ liệu tham chiếu mà còn là nền tảng quan trọng thúc đẩy nghiên cứu và triển khai bảo mật IoT trong tương lai.

2.2.2. Cấu trúc tập dữ liệu

a, Số lượng mẫu

Tập dữ liệu CiCIoT2023 bao gồm các dữ liệu lành tính và lưu lượng mạng độc hại. Tập dữ liệu có tổng 46,686,579 lưu lượng, trong đó có 1,098,195 lưu lượng lành tính và 45,588,384 lưu lượng độc hại. Số lượng mẫu theo từng nhóm và chi tiết từng loại tấn công thể hiện chi tiết tại bảng 2.1.

Mỗi loại tấn công mạng đều tạo ra một kiểu hành vi riêng biệt trong lưu lượng dữ liệu. Những khác biệt này không chỉ thể hiện ở kích thước lưu lượng mà còn phản ánh trong nhiều thuộc tính khác của tập dữ liệu. Tập dữ liệu lớn, tạo áp lực rất lớn lên mô hình học máy được áp dụng. Để giảm gánh nặng, đề án sử dụng tập dữ liệu con với tỷ lệ 1% tập gốc (465.000 mẫu) để đưa vào huấn luyện và đánh giá mô hình.

Bảng 2. 1 Phân bổ tấn công trong tập dữ liệu CICIOT2023

Loại tấn công	Số mẫu	Loại tấn công	Số mẫu	Nhóm tấn công	Số mẫu
DDoS-ICMP_Flood	7,200,504	DoS-TCP_Flood	2,671,445	DDoS	33,984,560
DDoS-UDP_Flood	5,412,287	DoS-SYN_Flood	2,028,834	DoS	8,090,738
DDoS-TCP_Flood	4,497,667	BenignTraffic	1,098,195	Mirai	2,634,124
DDoS-PSHACK_Flood	4,094,755	Mirai-greeth_flood	991,866	Benign	1,098,195
DDoS-SYN_Flood	4,059,190	Mirai-udpplain	890,576	Spoofing	486,504
DDoS-RSTFINFlood	4,045,285	Mirai-greip_flood	751,682	Recon	354,565
DDoS-SynonymousIP_Flood	3,598,138	DDoS-ICMP_Fragmentation	452,489	Web	24,829
DoS-UDP_Flood	3,318,595	MITM-ArpSpoofing	307,593	BruteForce	13,064
Recon-PingSweep	2262	Uploading_Attack	1252	Total	46,686,579
DDoS-UDP_Fragmentation	286,925	DDoS-HTTP_Flood	28,790		
DDoS-ACK_Fragmentation	285,104	DDoS-SlowLoris	23,426		
DNS_Spoofing	178,911	DictionaryBruteForce	13,064		
Recon-HostDiscovery	134,378	BrowserHijacking	5859		
Recon-OSScan	98,259	CommandInjection	5409		
Recon-PortScan	82,284	SqlInjection	5245		
DoS-HTTP_Flood	71,864	XSS	3846		
VulnerabilityScan	37,382	Backdoor_Malware	3218		

b, Các đặc trưng

Tập dữ liệu CICIoT2023 bao gồm 47 đặc trưng mô tả hành vi của luồng mạng (network flow) trong môi trường IoT. Các đặc trưng này được xây dựng nhằm phản ánh các khía cạnh quan trọng của lưu lượng mạng, hỗ trợ quá trình nhận diện và phân tích các hành vi bất thường hoặc tấn công mạng. Cụ thể, các đặc trưng có thể được phân loại theo các nhóm nội dung như sau:

- Đặc trưng liên quan đến thời gian và cấu trúc luồng dữ liệu bao gồm dấu thời gian khởi phát gói tin, thời lượng luồng và khoảng cách thời gian giữa các gói. Các thông số này phản ánh nhịp độ truyền dữ liệu và được sử dụng để phát hiện các hình thái tấn công gây đột biến về tải hoặc tần suất truy cập.

- Đặc trưng thống kê về kích thước gói tin như tổng kích thước, giá trị cực trị, trung bình và độ lệch chuẩn giúp mô tả sự phân bố dữ liệu trong một luồng. Những

biến động lớn về độ dài gói tin có thể gợi ý sự xuất hiện của các tấn công thao túng dữ liệu hoặc truyền tải thông tin bất thường.

- Đặc trưng liên quan đến tốc độ truyền dữ liệu (Rate, Srate, Drate) cung cấp thông tin về lưu lượng truyền tải theo từng hướng đi và đến. Đây là các chỉ báo quan trọng giúp nhận diện các cuộc tấn công DoS/DDoS vốn khai thác việc tạo ra lưu lượng lớn trong thời gian ngắn.

- Đặc trưng mô tả loại giao thức ở nhiều lớp (liên kết dữ liệu, mạng, vận chuyển, ứng dụng) được sử dụng dưới dạng biến nhị phân, cho phép phân biệt rõ ràng giữa các kiểu traffic hợp lệ và các hình thức khai thác giao thức bất thường, ví dụ như tấn công dựa trên HTTP, DNS hoặc ICMP.

- Đặc trưng về các cờ điều khiển TCP (FIN, SYN, RST, ACK, PSH, ECE, CWR) thể hiện hành vi điều khiển luồng và được xem là chỉ báo mạnh trong việc phát hiện các kỹ thuật tấn công như quét cổng, SYN flood, hoặc tấn công từ chối dịch vụ ở tầng vận chuyển.

- Đặc trưng nâng cao như Magnitude, Radius, Covariance và Variance mô tả mối quan hệ giữa dòng dữ liệu đi và đến trong một luồng, phản ánh tính đối xứng và mức độ ổn định của giao tiếp mạng. Các đặc trưng này hỗ trợ mô hình phân biệt các loại tấn công đơn hướng (như UDP flood) so với các cuộc tấn công có tương tác hai chiều.

2.2.3. Các loại tấn công trong tập dữ liệu

Bộ dữ liệu chứa tất cả 34 loại tấn công khác nhau nhằm vào các thành phần và giao thức phổ biến trong môi trường IoT. Các loại tấn công này có kịch bản thực tế cáo, là các loại tấn công đã biết trong môi trường IoT. Đề án phân tích phương pháp tối ưu phân loại trên 33 loại tấn công thay vì tập trung vào các loại tấn công có độ nhận diện cao. Bảng 2.3 thể hiện đầy đủ 33 cuộc tấn công phân vào 7 nhóm chính được thực hiện trong phòng thí nghiệm.

Bảng 2. 2 Các loại tấn công trong tập dữ liệu CICIOT2023

Lớp tấn công	Kịch bản tấn công	Cách thức tấn công
DoS	DoS-UDP_Flood	Kẻ tấn công gửi ồ ạt các gói UDP đến mục tiêu, làm cạn tài nguyên và ngăn truy cập hợp pháp.

	DoS-SYN_Flood	Gửi nhiều gói SYN để tạo kết nối nửa chừng trong TCP, làm đầy bảng kết nối của máy chủ.
	DoS-TCP_Flood	Gửi nhiều gói TCP ngẫu nhiên để làm nghẽn băng thông và giảm hiệu suất dịch vụ.
	DoS-HTTP_Flood	Tấn công ở tầng ứng dụng bằng cách gửi nhiều yêu cầu HTTP hợp lệ nhằm làm quá tải máy chủ web.
DDoS	DDoS-HTTP_Flood	Giống HTTP_Flood nhưng thực hiện từ nhiều nguồn zombie/botnet, mạnh hơn, khó đối phó hơn.
	DDoS-SlowLoris	Gửi yêu cầu HTTP rất chậm, giữ các kết nối mở để chiếm tài nguyên webserver.
	DDoS-ICMP_Fragmentation	Gửi nhiều gói ICMP bị phân mảnh khiến mục tiêu tốn tài nguyên xử lý và tái lắp ghép.
	DDoS-UDP_Fragmentation	Tương tự nhưng với giao thức UDP, làm hệ thống overload khi phải xử lý phân mảnh gói tin.
	DDoS-ACK_Fragmentation	Flood gói ACK bị phân mảnh làm nghẽn xử lý ở tầng TCP của mục tiêu.
	DDoS-SynonymousIP_Flood	Nhiều gói đến từ các địa chỉ IP đồng nghĩa/giả mạo khiến hệ thống khó lọc và quá tải.
	DDoS-ICMP_Flood	Flood ICMP (ping) khiến máy chủ phải liên tục phản hồi, làm suy giảm dịch vụ.
	DDoS-TCP_Flood	Flood các gói TCP từ nhiều nguồn, kết hợp phân tán gây ảnh hưởng lớn hơn DoS.
	DDoS-UDP_Flood	UDP Flood từ nhiều nút botnet, làm nghẽn mạng nhanh và mạnh.
	DDoS-SYN_Flood	SYN Flood phân tán, cực kỳ nguy hiểm cho hệ thống TCP handshake.
	DDoS-PSHACK_Flood	Gửi nhiều gói TCP PSH+ACK nhằm làm cạn tài nguyên xử lý dữ liệu đẩy/push.
	DDoS-RSTFINFlood	Flood gói RST/FIN để liên tục đóng kết nối, gây rối xử lý giao tiếp của mục tiêu.
Web-based	CommandInjection	Chèn lệnh hệ thống vào tham số đầu vào để chiếm quyền điều khiển server.
	SqlInjection	Chèn lệnh SQL độc hại nhằm đánh cắp hoặc thay đổi dữ liệu trong DB.
	Uploading_Attack	Tải lên file độc hại (webshell, mã độc) để chiếm quyền hoặc phá hoại hệ thống.
	XSS	Chèn script vào trang web để đánh cắp cookie, chiếm phiên hoặc điều khiển trình duyệt.

	Backdoor_Malware	Mã độc cài đặt cửa hậu trên thiết bị để truy cập và điều khiển trái phép.
	BrowserHijacking	Chiếm quyền điều hướng trình duyệt, chuyển hướng đến trang độc hại hoặc thu thập dữ liệu.
Recon	Recon-HostDiscovery	Quét tìm thiết bị đang hoạt động trên mạng để chuẩn bị tấn công.
	VulnerabilityScan	Quét lỗ hổng hệ thống nhằm tìm điểm yếu khai thác tiếp theo.
	Recon-PortScan	Quét cổng dịch vụ để xác định dịch vụ đang chạy và chuẩn bị khai thác.
	Recon-OSScan	Dò hệ điều hành mục tiêu bằng phân tích gói tin để lựa chọn kỹ thuật tấn công phù hợp.
	Recon-PingSweep	Gửi nhiều ping để xác định dải địa chỉ IP sống trong mạng.
Spoofing	MITM-ArpSpoofing	Giả mạo ARP để chen vào luồng dữ liệu, đánh cắp hoặc chỉnh sửa thông tin truyền.
	DNS_Spoofing	Giả mạo phản hồi DNS để chuyển hướng nạn nhân đến website độc hại.
Brute Force	DictionaryBruteForce	Dò mật khẩu bằng tập từ điển để chiếm quyền truy cập tài khoản dịch vụ.
Mirai	Mirai-greip_flood	Một biến thể của botnet Mirai flood gói GRE (Generic Routing Encapsulation) gây tắc nghẽn mạng.
	Mirai-udpplain	Flood các gói UDP đơn giản từ thiết bị IoT bị nhiễm Mirai.
	Mirai-greeth_flood	Flood gói GRE Ethernet — biến thể Mirai nguy hiểm ảnh hưởng lớn đến thiết bị mạng.

2.2.4. Tập dữ liệu CICIoT2023 trong nghiên cứu và đề án

Tập dữ liệu IoT2023 được lựa chọn là nguồn dữ liệu của đề án với mục đích đánh giá các mô hình học máy. Mục đó cuối cùng hướng tới hệ thống phát hiện xâm nhập để tối ưu nhằm tăng tính bảo mật trong môi trường Iot. Việc lựa chọn này có nhiều lý do:

Tập dữ liệu CICIoT2023 được xây dựng trong môi trường phòng thí nghiệm mô phỏng hệ sinh thái IoT, với mục đích phục vụ các nghiên cứu về an ninh và phát hiện tấn công mạng trên thiết bị IoT. Một điểm nổi bật của tập dữ liệu này là việc tích hợp nhiều kịch bản tấn công hiện đại và mang tính tổng quát cao, thúc đẩy các mô hình học máy học được các mẫu hành vi tấn công đa dạng và tăng khả năng thích ứng với nhiều bối cảnh thực tế khác nhau. Bên cạnh đó, CICIoT2023 cung cấp dữ liệu ở nhiều định dạng như PCAP và CSV, hỗ trợ linh hoạt, từ xử lý gói tin thô đến trích xuất đặc trưng mức cao. Điều này cho phép hệ thống và thuật toán được đánh giá trong nhiều điều kiện xử lý dữ liệu.

Tập dữ liệu này là quy mô lớn, với khoảng 46 triệu mẫu, phản ánh đặc trưng của môi trường IoT nơi lượng dữ liệu sinh ra liên tục và có tính thời gian thực. Quy mô này giúp mô phỏng sát hơn các hệ thống IoT lớn, hỗ trợ phát triển và đánh giá các mô hình có khả năng mở rộng, đảm bảo tính ổn định và hiệu quả khi triển khai trong môi trường hoạt động thực tế.

2.2.5. Tiền xử lý dữ liệu

Tiền xử là bước đầu tiên quan trọng nhất trong việc quá trình phân tích và đào tạo mô hình học máy. Việc tiền xử lý đảm bảo tính toàn vẹn, chính xác trước khi đưa vào huấn luyện mô hình học máy.

Mã hóa đặc trưng

Các mô hình học máy chỉ nhận và xử lý dữ liệu số. Vì vậy trước khi đưa dữ liệu vào huấn luyên, mã hóa dữ liệu là một bước quan trọng trong quá trình tiền xử lý dữ liệu. Mã hóa Label Encoder được sử dụng trong bước này. Phương pháp này ánh xạ dữ liệu phân loại từ dạng string sang dạng số nguyên.

Trong tập dữ liệu CICIoT2023, label encoder sẽ mã hóa dữ liệu của feature "Label" - nhãn các cuộc tấn công. Quá trình mã hóa được biểu diễn bởi bảng 2.4

Bảng 2. 3 Gán nhãn các loại tấn công

Nhãn mã hóa	Nhãn gốc	Nhãn mã hóa	Nhãn gốc
1	Backdoor_Malware	18	DictionaryBruteForce
2	BenignTraffic	19	DoS-HTTP_Flood
3	BrowserHijacking	20	DoS-SYN_Flood
4	CommandInjection	21	DoS-TCP_Flood
5	DDoS-ACK_Fragmentation	22	DoS-UDP_Flood
6	DDoS-HTTP_Flood	23	MITM-ArpSpoofing
7	DDoS-ICMP_Flood	24	Mirai-greeth_flood
8	DDoS-ICMP_Fragmentation	25	Mirai-greip_flood
9	DDoS-PSHACK_Flood	26	Mirai-udpplain
10	DDoS-RSTFINFlood	27	Recon-HostDiscovery
11	DDoS-SYN_Flood	28	Recon-OSScan
12	DDoS-SlowLoris	29	Recon-PingSweep
13	DDoS-SynonymousIP_Flood	30	Recon-PortScan
14	DDoS-TCP_Flood	31	SqlInjection
15	DDoS-UDP_Flood	32	Uploading_Attack
16	DDoS-UDP_Fragmentation	33	VulnerabilityScan
17	DNS_Spoofing	34	XSS

Quá trình Label Encoder mang lại nhiều lợi ích trong xử lý tập dữ liệu CICIoT2023, đặc biệt vì đây là bộ dữ liệu lớn và chứa nhiều dạng tấn công được mô tả bằng chuỗi. Label Encoder giúp chuyển đổi các nhãn tấn công từ dạng ký tự sang dạng số, tạo điều kiện để các mô hình học máy xử lý dữ liệu hiệu quả hơn. Việc mã hóa này cũng giúp tiết kiệm bộ nhớ và giảm độ phức tạp so với các phương pháp như

One-Hot Encoding, vốn tạo ra nhiều cột đặc trưng không cần thiết. Nhờ đó, quá trình huấn luyện mô hình trở nên nhanh hơn và ổn định hơn, đặc biệt với các thuật toán dựa trên cây như Random forest hoặc XGBoost. Ngoài ra, Label Encoder đảm bảo độ nhất quán khi phân loại nhiều loại tấn công IoT khác nhau, giúp mô hình hiểu và phân biệt chính xác giữa các lớp.

Chuẩn hóa dữ liệu

Phương pháp Standard Scaler, nhằm đưa tất cả các biến về cùng một phạm vi đo lường. "Chuẩn hóa là quá trình chuyển đổi giá trị của từng đặc trưng sao cho có trung bình bằng 0 và phương sai bằng 1", được biểu diễn theo công thức:

$$z = \frac{x - \mu}{\sigma} \qquad (2.11)$$

μ là giá trị trung bình và σ là độ lệch chuẩn của đặc trưng. Việc chuẩn hóa có vai trò quyết định trong học máy, bởi hầu hết các thuật toán phân loại và hồi quy hoạt động hiệu quả hơn và hội tụ nhanh hơn khi các đặc trưng đầu vào có cùng phạm vi giá trị. Nếu một đặc trưng có biên độ lớn hơn đáng kể so với các đặc trưng khác, nó có thể chi phối quá trình học và dẫn đến mô hình thiên lệch, làm giảm độ chính xác. Chuẩn hóa đảm bảo rằng mỗi đặc trưng đóng góp công bằng vào quá trình học, từ đó nâng cao khả năng mô hình nhận diện các mẫu và mối quan hệ tiềm ẩn một cách hiệu quả hơn.

Thông qua chuỗi các bước tiền xử lý dữ liệu bộ dữ liệu thô được chuyển đổi thành một định dạng sạch, đồng nhất và sẵn sàng cho các thuật toán học máy. Sau khi chuẩn hóa, dữ liệu được xử lý bằng các phương pháp đặc biệt để tăng hiệu suất và giảm thời như giảm chiều dữ liệu, phân cụm. Trong khuôn khổ đề án, phương pháp phân cụm được triển khai để cải thiện hiệu suất cho mô hình học máy Random forest, Decision tree.

2.5. Kết luận chương

Chương 2 đã trình bày chi tiết lý thuyết và kiến trúc tổng thể của hệ thống IDS dựa trên học máy trong môi trường IoT. Mục tiêu chính của hệ thống là nâng cao hiệu quả phát hiện tấn công, đồng thời đảm bảo khả năng triển khai trên các thiết bị biên

với tài nguyên hạn chế. Chương 2 đã trình bày nền tảng toán học của phương pháp phân cụm FCM, bao gồm hàm mục tiêu, cơ chế cập nhật trọng số và cách thức đo khoảng cách giữa các điểm dữ liệu. Những cơ sở này đóng vai trò quan trọng trong quy trình hỗ trợ mô hình học máy cải thiện hiệu năng phân loại.

Chương 2 đã làm rõ mối liên hệ giữa các thành phần trong hệ thống IDS và cung cấp đầy đủ nền tảng lý thuyết cần thiết cho quá trình triển khai, huấn luyện và đánh giá thực nghiệm trong Chương 3. Đây là cơ sở quan trọng để kiểm chứng hiệu quả và tính khả thi của mô hình đề xuất trên tập dữ liệu CICIoT2023

CHƯƠNG 3 - THỰC NGHIỆM VÀ ĐÁNH GIÁ KẾT QUẢ CỦA MÔ HÌNH IDS CHO MÔI TRƯỜNG IOT

3.1. Môi trường thực nghiệm

3.1.1. Cấu hình của mô hình huấn luyện

Đề án sử dụng môi trường lập trình dựa trên Python phiên bản 3.7. Python là ngôn ngữ lập trình bậc cao, được sử dụng nhiều trong lĩnh vực khoa học dữ liệu nhờ cú pháp rõ ràng, dễ tiếp cận và thư viện vô cùng phong phú. Quá trình được tiến hành trên hệ điều hành Windows 10 với các công cụ và thư viện chính như sau:

Python 3.7: Đây là phiên bản ổn định, tương thích tốt với nhiều thư viện học sâu, cho phép xử lý dữ liệu, xây dựng và quản lý thí nghiệm một cách linh hoạt. Python mang lại môi trường phát triển mạnh mẽ, hỗ trợ lập trình một cách nhanh và hiệu quả.

Keras: Keras hoạt động như một API cấp cao chạy trên TensorFlow, với mục tiêu đơn giản hóa quy trình phát triển mô hình học sâu. Thư viện này cung cấp các module được thiết kế sẵn, cho phép xây dựng mô hình theo dạng "lắp ghép", giúp tiết kiệm thời gian và giảm độ phức tạp trong lập trình. Nhờ tính trực quan của Keras, việc thiết kế kiến trúc mạng và tinh chỉnh tham số trở nên dễ dàng hơn rất nhiều.

Việc lựa chọn Python kết hợp Keras xuất phát từ những ưu thế vượt trội của bộ công cụ này đối với các bài toán học sâu. Python là ngôn ngữ chủ đạo trong học máy với cộng đồng hỗ trợ lớn. Keras giúp đơn giản quá trình phát triển nhưng vẫn đảm bảo sự linh hoạt. Nhờ đó, đề án có được một môi trường toàn diện, mạnh mẽ và tối ưu để nghiên cứu mô hình học máy.

3.1.2. Kịch bản thử nghiệm và đánh giá

Mục tiêu chính của đề án là phát triển hệ thống phát hiện xâm nhập (IDS) dựa trên sự kết hợp giữa phân cụm mờ Fuzzy C-Means (FCM) và mô hình học máy, nhằm tối ưu hóa khả năng nhận diện tấn công mạng trong môi trường IoT. Kịch bản thử nghiệm bao gồm bước như sau:

Tiền xử lý: Dữ liệu mạng sau khi thu thập được tiền xử lý nhằm đảm bảo tính nhất quán và chất lượng, bao gồm các bước chuẩn hóa dữ liệu, mã hóa thuộc tính dạng phân loại và chia dữ liệu thành các tập huấn luyện.

Phân cụm dữ liệu: Thuật toán FCM được áp dụng với khoảng cách Cosine và số lượng cụm được thay đổi từ 2 đến 9 nhằm khảo sát tác động của số cụm và khoảng cách đến khả năng học và phân biệt mẫu tấn công của hệ thống IDS.

Huấn luyện mô hình: Sau quá trình phân cụm, ma trận độ phụ thuộc về của FCM được sử dụng làm đầu vào cho các mô hình học máy, trong đó giá trị độ phụ thuộc về phản ánh mức độ gán mờ của từng mẫu vào các cụm; đây được xem là nguồn đặc trưng bổ sung giúp mô hình học máy học tốt hơn các ranh giới phức tạp của dữ liệu mạng. Mô hình học máy được triển khai độc lập cho từng cấu hình phân cụm gồm Random forest, Decision tree. Lý do cho việc này là mỗi cụm có thể chứa một tập hợp các loại tấn công (class) khác nhau. Việc huấn luyện một mô hình chuyên biệt cho từng cụm giúp tối ưu hóa khả năng nhận diện các lớp đặc thù trong cụm đó. Kết quả của giai đoạn này là một tập hợp gồm N mô hình phân loại đã được huấn luyện, tương ứng với N cụm dữ liệu.

Trong giai đoạn kiểm thử, mỗi mẫu dữ liệu được xác định khoảng cách đến tâm cụm đã được tính toán ở giai đoạn huấn luyện. Mẫu này sẽ được gán vào tâm cụm gần nhất. Sau đó thông qua mô hình phân loại đã được huấn luyện trước đó để đưa ra dự đoán về nhãn tấn công.

Đánh giá hiệu suất: Mỗi mô hình được huấn luyện trên dữ liệu có gắn nhãn kết hợp với đặc trưng phân cụm, sau đó được đánh giá trên tập kiểm định theo các thước đo hiệu năng gồm:Accuracy, Precision, Recall, F1-score

Kết quả thu được từ các cấu hình khoảng cách số cụm mô hình học máy được tổng hợp và so sánh nhằm xác định ảnh hưởng của từng yếu tố đến hiệu suất phát hiện tấn công. Cấu hình tối ưu cuối cùng được lựa chọn dựa trên khả năng đạt hiệu suất cao ổn định trên tập kiểm tra, đồng thời đảm bảo mức độ phức tạp tính toán phù hợp cho triển khai IDS trong môi trường IoT thực tế. Phân tích cuối cùng làm rõ lợi ích của việc tích hợp FCM vào hệ thống IDS, bao gồm khả năng cải thiện độ chính

xác nhờ khai thác thông tin mờ của dữ liệu và tăng cường khả năng mô hình hóa các mẫu tấn công có ranh giới không tuyến tính

3.2. Kết quả đạt được

3.2.1. Các tiêu chí đánh giá hiệu năng

a, Độ chính xác tổng thể (Accuracy)

Độ chính xác tổng thể (Accuracy) là một trong những chỉ số cơ bản trong đánh giá các mô hình phân loại. Chỉ số này được xác định bằng tỷ lệ giữa dự đoán chính xác của mô hình (bao gồm cả các trường hợp dự đoán đúng dương tính và đúng âm tính) so với tổng số mẫu được đưa vào kiểm tra.

Accuracy được biểu diễn bởi công thức:

$$Accuracy = \frac{TN + TP}{FP + FN + TP + TN} \tag{3.1}$$

- TP định nghĩa là số mẫu thuộc lớp dương tính được phân loại chính xác
- TN là mẫu thuộc lớp âm tính được phân loại chính xác
- FP là số mẫu lớp âm tính bị phân loại sai thành dương tính
- FN là số mẫu lớp dương tính bị phân loại sai thành âm tính

b, Độ phủ (Recall)

Recall, hay còn được gọi là "độ nhạy" hoặc "tỷ lệ phát hiện đúng", là một chỉ số đặc biệt quan trọng trong lĩnh vực an ninh mạng. Recall phản ánh mức độ mà hệ thống có khả năng phát hiện đầy đủ các mối đe dọa tiềm ẩn, do đó đóng vai trò then chốt trong việc bảo đảm an toàn cho hệ thống mạng.

$$Recall = \frac{TP}{TP + FN} \tag{3.2}$$

c, Độ chính xác (Precision)

Precision là chỉ số đo lường mức độ chính xác của các cảnh báo tấn công do hệ thống phát ra. Chỉ số này xác định bằng tỷ lệ giữa số lượng cảnh báo đúng (True Positives) và tổng số cảnh báo được hệ thống tạo ra, bao gồm cả các cảnh báo đúng và các cảnh báo sai (True Positives + False Positives). Precision cao cho thấy rằng các cảnh báo được phát hành có mức độ tin cậy lớn, tức là xác suất một cảnh báo phản ánh đúng một cuộc tấn công thực sự là rất cao. Điều này góp phần giảm thiểu

hiện tượng "mệt mỏi cảnh báo" (alert fatigue) mà các quản trị viên mạng thường gặp phải khi phải xử lý một số lượng lớn cảnh báo không chính xác.

$$Precision = \frac{TP}{TP + FP} \tag{3.3}$$

d, F1-score

F1-score là chỉ số được xác định dưới dạng trung bình điều hòa giữa Precision và Recall. Nhờ đặc tính kết hợp cả hai tiêu chí, F1-score mang lại một đánh giá cân bằng về hiệu quả của mô hình, phù hợp khi dữ liệu mất cân bằng, nơi một trong hai chỉ số riêng lẻ có thể gây hiểu nhầm về năng lực thực sự của mô hình.

$$F1 - score = \frac{2 \cdot (Precision \cdot Recall)}{Precision + Recall} \tag{3.4}$$

F1-score đạt giá trị tốt khi cả Precision và Recall đều được duy trì ở mức tốt, do đây là chỉ số phản ánh sự hài hòa giữa hai yếu tố này. Khi chỉ số F1-score ở mức cao, mô hình có thể được xem là hoạt động hiệu quả trên cả hai khía cạnh: nhận diện chính xác các mẫu dương tính và giảm thiểu số lượng dự đoán sai, qua đó nâng cao độ tin cậy tổng thể của hệ thống phát hiện xâm nhập.

3.2.2. Đánh giá mô hình học máy khi không dùng phân cụm FCM

Phần này đánh giá hiệu suất khi chỉ sử dụng mô hình học máy lên tập dữ liệu CICIoT2023. Đề án đề xuất hai mô hình học máy Random forest, Decision tree.

Bảng 3. 1 Độ chính xác của mô hình học máy trên tập dữ liệu CICIOT2023

Mô hình	RandomForest	Decision tree
Accuracy	0.9877	0.9523
Precision_Macro	0.7771	0.7072
Recall_Macro	0.7253	0.477
F1_Macro	0.7398	0.4878
Precision_Weighted	0.9865	0.9544
Recall_Weighted	0.9877	0.9523
F1_Weighted	0.9867	0.9459

Kết quả bảng 3.1 khi thực hiện huấn luyện mô hình học máy được đề xuất. Mô hình Random forest huấn luyện với tham số chính n_estimators. n_estimators được định nghĩa số cây quyết định được sử dụng trong mô hình học máy random forest. Số cây quyết định càng lớn mô hình càng ổn định nhưng đồng thời cũng tăng tính phức tạp và thời gian huấn luyện. Với mục tiêu thiết kế mô hình nhẹ, đề án sử dụng n_estimators = 10. Tham số này khá nhỏ, tuy nhiên kết quả huấn luyện của mô hình vẫn rất tốt với độ chính xác đạt 98,77%. Điều này chứng tỏ mô hình đã học rất tốt những đặc trưng của tập dữ liệu.

Bên cạnh đó, đề án sử dụng tham số F1-marco để tính trung bình F1-score trên tất cả các lớp với trọng số các lớp là ngang nhau, không phụ thuộc vào số lượng mẫu của mỗi lớp. Điều này cho phép F1-marco đánh giá hiệu suất mô hình trên tất cả các lớp, bao gồm cả các lớp tỷ lệ xuất hiện thấp. F1_marco đạt 73,93%. Tương tự chỉ số Precision_marco và recal_marco lần lượt đạt 77.77% và 72.53%, điều này chứng tỏ mô hình có khả năng phân loại với các lớp tấn công ít số liệu. Kết quả này chưa thật sự cao do số lượng mẫu còn hạn chế.

Tham số weight tính trung bình trên các lớp có tính đến tỷ lệ mẫu của từng lớp trong tập dữ liệu. Vì vậy tham số weight thể hiện hiệu suất tổng thể của mô hình học máy. Với Random forest, hiệu suất tổng thể vẫn giữ được ở mức rất cao (F1_weight = 98,67%, Recall_weight = 98,77%, Precision_weight = 98.65%)

Với mô hình học máy Decison Tree, với độ sâu max_depth = 10, kết quả tổng thể mặc dù thấp hơn so với mô hình Random forest nhưng vẫn đạt trên 90% (accuracy = 95,23%, F1_weight = 94.59%, Precision_weight= 95,44%, Recall_weight = 95,23%). Tuy nhiên F1_marco khá thấp, chỉ đạt 48,78%.

Từ kết quả trên có thể thấy, với tập dữ liệu bị mất cân bằng nặng như CICIoT2023, việc chỉ sử dụng mô hình học máy truyền thống chưa thật sự hiệu quả với các lớp có tỷ lệ xuất hiện thấp.

3.2.3. Đánh giá mô hình học máy khi sử dụng phương pháp phân cụm

Đề án đánhh giá phương pháp phân cụm FCM đã đề cập sử dụng khoảng cách cosine. Số cụm được thiết lập tăng dần để kiểm thử hiệu quả mô hình.

a, Mô hình học máy khi kết hợp phân cụm FCM

Bảng 3. 2 Kết quả mô hình Decision tree kết hợp phân cụm FCM

Số cụm	Accuracy	F1_ Macro	Precision_ Macro	Recall_ Macro	F1_ Weighted	Precision_ Weighted	Recall_ Weighted
2	0.9897	0.7392	0.8009	0.7236	0.9887	0.9891	0.9897
3	0.9889	0.8586	0.9235	0.8218	0.9885	0.9889	0.9889
4	0.9914	0.8679	0.9278	0.8334	0.9910	0.9913	0.9914
5	0.9914	0.8414	0.8912	0.8185	0.9911	0.9915	0.9914
6	0.9908	0.8496	0.9040	0.8265	0.9906	0.9912	0.9908
7	0.9906	0.8346	0.8854	0.8160	0.9904	0.9912	0.9906
8	0.9886	0.7961	0.8530	0.7786	0.9885	0.9894	0.9886
9	0.9901	0.8298	0.8727	0.8203	0.9900	0.9906	0.9901

Kết quả bảng 3.2 khi thực hiện phân cụm 2; 3; 4; 5; 6; 7; 8; 9 cho thấy kết quả của mô hình Decision tree có hiệu suất rất tốt. Độ chính xác mô hình gần như đạt mức tuyệt đối. Điều này cho thấy hiệu quả của phân cụm trong việc nâng cao hiệu quả mô hình. Đối với mô hình Decision tree, việc lựa chọn số cụm K trong giai đoạn phân cụm có ảnh hưởng trực tiếp đến hiệu quả phân loại. Dựa trên kết quả thực nghiệm, mô hình đạt hiệu năng tối ưu khi số cụm được thiết lập là K=4, thể hiện rõ qua các chỉ số đánh giá, đặc biệt là F1-macro. Cụ thể, độ chính xác tổng thể đạt khoảng 99%, trong khi F1-macro đạt 86,79%, cho thấy sự cân bằng tốt giữa độ chính xác và khả năng phân biệt các lớp.Giá trị F1-macro cao chứng minh mô hình không chỉ tối ưu cho các lớp có tần suất xuất hiện lớn mà còn học được đặc trưng của các lớp tấn công hiếm, vốn thường bị bỏ qua trong các bài toán dữ liệu mất cân bằng. Việc phân cụm với K=4 giúp dữ liệu được phân tách hợp lý theo các đặc trưng hành vi, tạo điều kiện để mô hình Decision tree xây dựng các ranh giới phân loại rõ ràng hơn cho từng nhóm lớp. Các chỉ số đánh giá khác cũng đạt giá trị cao và ổn định, qua đó khẳng định K=4 là lựa chọn phù hợp nhất cho mô hình trong nghiên cứu này.

Kết quả bảng 3.3 cho thấy mô hình học máy Random forest đạt kết quả khá tốt khi kết hợp với phương pháp phân cụm dữ liệu. Đặc biệt với k=8, mô hình cho kết quả tối ưu với độ chính xác accuracy đạt 98,79%. Tham số Marco cũng đạt được

kết quả khá tốt (F1_macro = 74.99%, Precision_macro = 82,82%, Recal_macro = 72,37%). Điều này chứng minh mô hình phân loại khá tốt lớp tấn công hiếm.

Khi tích hợp phương pháp phân cụm vào quá trình học máy, hiệu suất của mô hình Random forest không còn vượt trội như trong cách tiếp cận truyền thống.

Bảng 3. 3 Kết quả mô hình Random forest kết hợp phân cụm FCM

Số Cụm	Accuracy	F1_ Macro	Precision _Macro	Recall_ Macro	F1_ Weighted	Precision_ Weighted	Recall_ Weighted
2	0.9881	0.7363	0.7931	0.7193	0.9869	0.9868	0.9881
3	0.9832	0.7107	0.8083	0.6885	0.9813	0.9830	0.9832
4	0.9868	0.7221	0.8306	0.6982	0.9851	0.9855	0.9868
5	0.9879	0.7332	0.8192	0.7146	0.9866	0.9869	0.9879
6	0.9873	0.7215	0.8582	0.7039	0.9858	0.9871	0.9873
7	0.9878	0.7270	0.7967	0.7149	0.9866	0.9870	0.9878
8	0.9879	0.7499	0.8282	0.7237	0.9867	0.9867	0.9879
9	0.9884	0.7444	0.8353	0.7231	0.9872	0.9876	0.9884

Ngược lại, mô hình Decision tree lại cho thấy khả năng phân loại và phát hiện tấn công tốt hơn trong bối cảnh này. Điều này chứng minh việc kết hợp phân cụm có xu hướng phù hợp hơn với các mô hình có cấu trúc đơn giản như Decision tree, do chúng dễ dàng thích ứng với thông tin nhóm được tạo ra từ giai đoạn phân cụm.

b, So sánh chỉ số hiệu suất mô hình học máy kết hợp phương pháp phân cụm và mô hình học máy truyền thống

- Chỉ số hiệu suất của mô hình học máy

Kết quả bảng 3.4 cho thấy phương pháp phân cụm đã tạo ra tác động tích cực đến hiệu suất của mô hình học máy.

Mô hình Decision tree khi kết hợp với phân cụm FCM (K=4) cho kết quả tối ưu hơn mô hình truyền thống ở tất cả các chỉ số đánh giá. Accuracy của mô hình tăng từ 95.23% lên 99.14%, trong khi F1-Macro tăng từ 48.78% lên 86.79%. Điều này chứng tỏ phương pháp phân cụm đã hỗ trợ đáng kể mô hình trong việc phân biệt tốt hơn các lớp tấn công có số mẫu nhỏ, vốn là thách thức chính trong bài toán phát hiện xâm nhập với dữ liệu mất cân bằng. Ngoài ra, sự gia tăng mạnh ở Recall-Macro và Precision-Macro cho thấy mô hình không chỉ giảm tỉ lệ bỏ sót các tấn công nguy hiểm mà còn hạn chế cảnh báo sai một cách hiệu quả.

Mặc dù hiệu quả của phương pháp phân cụm không cao như Decison tree, mô hình Random forest vẫn có sự cải thiện nhất định khi kết hợp với FCM (K=8). Hầu hết các chỉ số như Accuracy, Precision_Weighted và F1_Weighted đều tăng nhẹ.

Bảng 3. 4. So sánh hiệu suất mô hình học máy truyền thông và mô hình học máy kết hợp phân cụm FCM

Mô hình	Decision tree		RandomForest	
	Mô hình truyền thống	Mô hình kế hợp phân cụm (K =4)	Mô hình truyền thống	Mô hình kết hợp phân cụm (K =8)
Accuracy	0.9523	0.9914	0.9877	0.9879
F1_Macro	0.4878	0.8679	0.7398	0.7499
Precision_Macro	0.7072	0.8912	0.7771	0.8282
Recall_Macro	0.477	0.8185	0.7253	0.7237
F1_Weighted	0.9459	0.9911	0.9867	0.9867
Precision_Weighted	0.9544	0.9915	0.9865	0.9867
Recall_Weighted	0.9523	0.9914	0.9877	0.9879

Điều này cho thấy phương pháp phân cụm vẫn có hiệu quả trong việc cải thiện hiệu suất mô hình. Tuy nhiên, Random forest vốn đã có khả năng tổng quát hóa tốt và ít chịu ảnh hưởng bởi nhiễu trong dữ liệu gốc, nên việc hiệu quả của phương pháp phân cụm không thể hiện rõ ràng.

- Độ chính xác theo từng lớp tấn công

Kết quả cụ thể cho việc phân loại tấn công được trình bày trong hình 3.2, hình bảng 3.5. Bảng này đã so sánh kết quả mô hình học máy Decision tree khi kết hợp với phương pháp phân cụm dữ liệu và mô hình truyền thống. Có thể thấy, mô hình truyền thống cho kết quả phân loại chưa tốt, nhiều lớp tấn công có độ chính xác thấp. Chứng tỏ mô hình chưa học tốt các đặc trưng cảu từng lớp tấn công. Bên cạnh đó, mô hình học máy khi kết hợp FCM đã cho kết quả rất khả quan. Hình 3.3 đã thể hiện kết quả đó. Hầu hết các lớp đều đạt mức tối ưu 0,9 đến 1 (14 lớp đạt mức tối ưu 1, 4 lớp đạt mức tối ưu 0.9). Các lớp còn lại đều đạt mức tối ưu tương đối. Một số lớp có mức tối ưu dưới mức trung bình có số mẫu quá nhỏ để mô hình có thể học được đặc trưng. Tỷ lệ rất nhỏ so với tổng số lượng mẫu nên có thể chấp nhận được. Bảng 3.6 so sánh hiệu suất giữa mô hình Random forest truyền thống và mô hình Random forest kết

hợp phương pháp phân cụm FCM. Từ bảng so sánh có thể thấy, mô hình truyền thống đã có khả năng phân loại đã đạt mức cao: phần lớn các mẫu được dự đoán chính xác tập trung trên đường chéo chính, trong đó có 9 lớp đạt độ chính xác tối (1.0) và 8 lớp đạt độ chính xác trên 0.9.

Bảng 3. 5 Báo cáo phân loại theo lớp tấn công mô hình Decison Tree

Tên lớp tấn công	Decision tree kết hợp phân cụm FCM (k=4)			Decision tree truyền thống			Số mẫu
	Precision	Recall	F1-score	Precision	Recall	F1-score	
Backdoor_Malware	0.81	0.53	0.64	0.33	0.02	0.04	47
BenignTraffic	0.88	0.94	0.91	0.34	0.98	0.51	2172
BrowserHijacking	0.91	0.62	0.73	0.40	0.04	0.08	47
CommandInjection	1.00	0.38	0.55	0.00	0.00	0.00	47
DDoS-ACK_ Fragmentation	0.99	0.99	0.99	0.00	0.00	0.00	548
DDoS-HTTP_Flood	0.90	0.98	0.94	1.00	0.11	0.19	47
DDoS-ICMP_Flood	1.00	1.00	1.00	1.00	1.00	1.00	14378
DDoS-ICMP_Fragmentation	0.94	1.00	0.97	0.00	0.00	0.00	883
DDoS-PSHACK_Flood	1.00	1.00	1.00	0.99	1.00	0.99	8166
DDoS-RSTFINFlood	1.00	1.00	1.00	1.00	1.00	1.00	8068
DDoS-SYN_Flood	1.00	1.00	1.00	1.00	1.00	1.00	8096
DDoS-SlowLoris	1.00	0.89	0.94	0.62	0.17	0.27	47
DDoS-SynonymousIP_Flood	1.00	1.00	1.00	1.00	1.00	1.00	7174

Tên lớp tấn công	Decision tree kết hợp phân cụm FCM (k=4)			Decision tree truyền thống			Số mẫu
	Precision	Recall	F1-score	Precision	Recall	F1-score	
DDoS-TCP_Flood	1.00	1.00	1.00	1.00	0.97	0.98	8972
DDoS-UDP_Flood	1.00	1.00	1.00	1.00	0.99	1.00	10800
DDoS-UDP_Fragmentation	0.98	1.00	0.99	0.00	0.00	0.00	552
DNS_Spoofing	0.74	0.68	0.71	0.86	0.05	0.10	332
DictionaryBruteForce	0.86	0.51	0.64	1.00	0.04	0.08	47
DoS-HTTP_Flood	1.00	1.00	1.00	1.00	0.91	0.95	126
DoS-SYN_Flood	0.99	1.00	0.99	1.00	0.98	0.99	4036
DoS-TCP_Flood	1.00	0.99	1.00	1.00	1.00	1.00	5319
DoS-UDP_Flood	0.99	1.00	1.00	1.00	1.00	1.00	6614
MITM-ArpSpoofing	0.76	0.81	0.79	0.78	0.32	0.45	593
Mirai-greeth_flood	1.00	1.00	1.00	1.00	0.99	1.00	1962
Mirai-greip_flood	1.00	1.00	1.00	1.00	0.98	0.99	1480
Mirai-udpplain	0.99	1.00	1.00	0.96	0.93	0.94	1760
Recon-HostDiscovery	0.96	0.73	0.83	0.52	0.44	0.48	240
Recon-OSScan	0.85	0.61	0.71	0.59	0.11	0.18	179
Recon-PingSweep	0.85	0.49	0.62	1.00	0.02	0.04	47
Recon-PortScan	0.75	0.56	0.64	0.67	0.10	0.17	140
SqlInjection	0.67	0.64	0.65	1.00	0.02	0.04	47
Uploading_Attack	0.85	0.47	0.60	0.00	0.00	0.00	47
VulnerabilityScan	0.98	0.91	0.95	0.00	0.00	0.00	47
XSS	0.90	0.60	0.72	1.00	0.06	0.12	47

Tên lớp tấn công	Decision tree kết hợp phân cụm FCM (k=4)			Decision tree truyền thống			Số mẫu
	Precision	Recall	F1-score	Precision	Recall	F1-score	
accuracy			0.99			0.95	93107
macro avg	0.927	0.83	0.86	0.95	0.47	0.48	93107
weighted avg	0.99	0.99	0.99	0.95	0.95	0.94	93107

Bảng 3. 6 Báo cáo phân loại theo lớp của mô hình Random forest

Tên lớp tấn công	Random forest kết hợp phân cụm FCM (K=8)			Random forest truyền thống			Số mẫu
	Precision	Recall	F1-score	Precision	Recall	F1-score	
Backdoor_Malware	0.38	0.17	0.24	0.26	0.28	0.27	47
BenignTraffic	0.86	0.97	0.91	0.81	0.97	0.89	2172
BrowserHijacking	0.71	0.26	0.38	0.37	0.28	0.32	47
CommandInjection	0.31	0.11	0.16	0.38	0.19	0.25	47
DDoS-ACK_Fragmentation	0.99	0.98	0.99	0.99	0.98	0.98	548
DDoS-HTTP_Flood	0.93	0.87	0.90	0.87	0.98	0.92	47
DDoS-ICMP_Flood	1.00	1.00	1.00	1.00	1.00	1.00	14378

Tên lớp tấn công	Random forest kết hợp phân cụm FCM (K=8)			Random forest truyền thống			Số mẫu
	Precision	Recall	F1-score	Precision	Recall	F1-score	
DDoS-ICMP_Fragmentation	0.98	0.98	0.98	0.99	0.98	0.99	883
DDoS-PSHACK_Flood	1.00	1.00	1.00	1.00	1.00	1.00	8166
DDoS-RSTFINFlood	1.00	1.00	1.00	1.00	1.00	1.00	8068
DDoS-SYN_Flood	0.99	1.00	1.00	1.00	1.00	1.00	8096
DDoS-SlowLoris	0.71	0.77	0.73	0.80	0.83	0.81	47
DDoS-SynonymousIP_Flood	1.00	0.99	1.00	1.00	1.00	1.00	7174
DDoS-TCP_Flood	1.00	1.00	1.00	1.00	1.00	1.00	8972
DDoS-UDP_Flood	1.00	1.00	1.00	1.00	1.00	1.00	10800
DDoS-UDP_Fragmentation	0.99	0.99	0.99	0.99	0.99	0.99	552
DNS_Spoofing	0.58	0.62	0.60	0.65	0.60	0.63	332
DictionaryBruteForce	0.57	0.09	0.15	0.36	0.11	0.16	47

Tên lớp tấn công	Random forest kết hợp phân cụm FCM (K=8)			Random forest truyền thống			Số mẫu
	Precision	Recall	F1-score	Precision	Recall	F1-score	
DoS-HTTP_Flood	0.92	0.98	0.95	0.95	0.96	0.96	126
DoS-SYN_Flood	1.00	1.00	1.00	1.00	1.00	1.00	4036
DoS-TCP_Flood	0.99	1.00	0.99	1.00	1.00	1.00	5319
DoS-UDP_Flood	0.98	1.00	0.99	1.00	1.00	1.00	6614
MITM-ArpSpoofing	0.83	0.78	0.80	0.81	0.75	0.78	593
Mirai-greeth_flood	1.00	1.00	1.00	1.00	1.00	1.00	1962
Mirai-greip_flood	0.99	0.99	0.99	1.00	0.99	0.99	1480
Mirai-udpplain	1.00	0.99	1.00	1.00	0.99	1.00	1760
Recon-HostDiscovery	0.82	0.68	0.74	0.70	0.65	0.68	240
Recon-OSScan	0.63	0.41	0.50	0.58	0.35	0.44	179
Recon-PingSweep	0.56	0.21	0.31	0.45	0.21	0.29	47
Recon-PortScan	0.57	0.39	0.46	0.58	0.37	0.45	140
SqlInjection	0.86	0.26	0.39	0.21	0.09	0.12	47
Uploading_Attack	0.73	0.17	0.28	0.47	0.17	0.25	47
VulnerabilityScan	0.93	0.81	0.86	0.87	0.87	0.87	47

Tên lớp tấn công	Random forest kết hợp phân cụm FCM (K=8)			Random forest truyền thống			Số mẫu
	Precision	Recall	F1-score	Precision	Recall	F1-score	
XSS	0.36	0.17	0.23	0.33	0.09	0.14	47
accuracy		0.98			0.98		93107
macro avg	0.82	0.72	0.74	0.77	0.7	0.73	93107
weighted avg	0.98	0.98	0.98	0.98	1	0.98	93107

Khi sử dụng phương pháp phân cụm FCM, mô hình cho thấy sự cải thiện đáng đáng kể. Số lớp đạt độ chính xác tối ưu tăng lên 11 lớp, và các lớp đạt trên mức 0.9 tăng lên 9 lớp. Sự gia tăng này cho thấy việc phân cụm đã góp phần giảm bớt sự chồng lấn giữa các không gian đặc trưng của các loại tấn công, từ đó cải thiện ranh giới quyết định của mô hình trong giai đoạn học. Mặc dù mức cải thiện không quá đột phá, nhưng kết quả cũng chứng minh rằng việc kết hợp phân cụm FCM mang lại hiệu quả đối với Random forest một mô hình học máy vốn đã mạnh về khả năng tổng quát hóa.

Kết quả chứng minh tính hiệu quả của mô hình, đặc biệt với các tập dữ liệu lớn với nhiều tham số như tập dữ liệu CICIoT2023. Từ kết quả phân tích, có thể thấy mô hình tối ưu nhất là mô hình học máy Decision tree kết hợp phương pháp phân cụm FCM.

3.3. Kết luận chương 3

Trong chương này, mô hình IDS dựa trên học máy kết hợp phân cụm FCM đã được triển khai, thử nghiệm và đánh giá trên tập dữ liệu CICIoT2023. Mô hình IDS được thiết kế với khả năng mở rộng cao nhờ kiến trúc phân tán, cho phép triển khai độc lập các mô-đun phát hiện trên từng thiết bị biên. Khi quy mô hệ thống IoT tăng lên, các nút IDS mới có thể được bổ sung mà không làm ảnh hưởng đến hiệu năng tổng thể của hệ thống. Bên cạnh đó, việc sử dụng các mô hình học máy nhẹ giúp duy trì tính ổn định và hiệu quả xử lý ngay cả khi số lượng thiết bị và lưu lượng dữ liệu

tăng đáng kể. Kết quả thực nghiệm cho thấy việc tích hợp phân cụm trước khi phân lớp giúp cải thiện hiệu năng phát hiện xâm nhập, đặc biệt trong bối cảnh dữ liệu mất cân bằng đặc trưng của môi trường IoT. Qua phân tích các chỉ số đánh giá, mô hình học máy decision tree kết hợp phương pháp phân cụm với số cụm k=4 là cấu hình tối ưu, duy trì độ chính xác 99,14%, đặc biệt F1_Macro đạt 86.79%. Điều này chứng minh rằng kỹ thuật phân cụm giúp tăng cường nhận diện các mẫu bất thường, giảm tỷ lệ dự đoán sai và tối ưu hóa khả năng học của mô hình. Như vậy, kết quả trong chương 3 đã xác thực tính khả thi và hiệu quả của mô hình IDS đề xuất khi áp dụng trong môi trường IoT thực tế

KẾT LUẬN VÀ KIẾN NGHỊ

Trong bối cảnh các hệ thống IoT và IIoT ngày càng mở rộng, nguy cơ tấn công mạng trở nên đa dạng và tinh vi hơn, đòi hỏi sự phát triển của các hệ thống phát hiện xâm nhập (IDS) có hiệu năng cao, độ tin cậy lớn và khả năng triển khai trên môi trường tính toán biên hạn chế tài nguyên. Đề án đã tập trung nghiên cứu, đề xuất và đánh giá mô hình IDS kết hợp học máy và phân cụm nhằm nâng cao hiệu quả phát hiện bất thường trên tập dữ liệu CICIoT2023.

Trên cơ sở phân tích dữ liệu thực nghiệm, mô hình phân cụm FCM đã chứng minh được tính khả thi trong việc xử lý dữ liệu mất cân bằng đặc trưng phổ biến trong an ninh IoT. Đặc biệt, mô hình học máy Decision tree kết hợp phương pháp phân cụm với số cụm tối ưu K=4 được xác định là tối ưu, giúp cải thiện độ chính xác, giảm lỗi nhận diện và tăng cường khả năng phát hiện các mẫu tấn công ẩn giấu. Kết quả này cho thấy kỹ thuật phân cụm đóng vai trò quan trọng trong bước tiền xử lý, góp phần tối ưu hóa việc học của mô hình và tăng hiệu năng của IDS trên môi trường IoT.

Nghiên cứu đã cung cấp đánh giá toàn diện về mô hình, chỉ ra tính hiệu quả của giải pháp đề xuất khi so sánh với mô hình gốc không sử dụng phân cụm. Qua đó khẳng định rằng phương pháp phân cụm là hướng tiếp cận phù hợp trong phát triển các hệ thống IDS thế hệ mới cho IoT..

Trong tương lai, nghiên cứu sẽ tiếp tục mở rộng theo các định hướng sau:

- Tối ưu hóa thuật toán phân cụm nhằm giảm chi phí tính toán và độ trễ, phục vụ triển khai thực tế trên thiết bị biên.
- Kết hợp phân cụm với các cơ chế học nhẹ, đảm bảo khả năng cập nhật mô hình liên tục mà không gây quá tải tài nguyên.
- Kết hợp với phương pháp giảm chiều đề giảm số đặc trưng, tăng tốc độ tính toán.
- Nghiên cứu mô hình học phân tán và ra quyết định hợp nhất từ nhiều thiết bị IoT, tối ưu hóa xử lý dữ liệu ở cấp độ mạng.
- Tiếp tục thử nghiệm trên các bộ dữ liệu lớn và kịch bản thực tế trong IoT để đánh giá khả năng tổng quát của mô hình.

DANH MỤC TÀI LIỆU THAM KHẢO

[1] M. Hatton, “The IoT in 2030: Which applications account for the biggest chunk of the $1.5 trillion opportunity?” [Online].

[2] Christos Xenofontos, Ioannis Zografopoulos, Charalambos Konstantinou, A. Jolfaei, M. Khan and K. Choo. "Consumer, Commercial, and Industrial IoT (In)Security: Attack Taxonomy and Case Studies." IEEE Internet of Things Journal, 9 (2021): 199-221. https://doi.org/10.1109/jiot.2021.3079916.

[3] Gartner, Inc, “Gartner Says 5.8 Billion Enterprise and Automotive IoT Endpoints Will Be in Use in 2020,” [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2019-08-29-gartner-says-5-8-billion-enterprise-and-automotive-io,2019.

[4] J. E. I. Esquer, F. F. González-Navarro, Brenda Leticia Flores Ríos, L. Burtseva and María Angélica Astorga Vargas. "Tracking the Evolution of the Internet of Things Concept Across Different Application Domains." Sensors (Basel, Switzerland), 17 (2017). https://doi.org/10.3390/s17061379

[5] Zhou, Ian, Imran Makhdoom, Negin Shariati, Muhammad Ahmad Raza, Rasool Keshavarz, Justin Lipman, Mehran Abolhasan, and Abbas Jamalipour. "Internet of things 2.0: Concepts, applications, and future directions." IEEE Access 9 (2021): 70961-71012.

[6] Sasaki, Y. (2022). A Survey on IoT Big Data Analytic Systems: Current and Future. IEEE Internet of Things Journal, 9, 1024-1036. https://doi.org/10.1109/jiot.2021.3131724.

[7] Ansam Khraisat, I. Gondal, P. Vamplew and J. Kamruzzaman. "Survey of intrusion detection systems: techniques, datasets and challenges." Cybersecurity, 2 (2019). https://doi.org/10.1186/s42400-019-0038-7.

[8] E. P. Neto, Sajjad Dadkhah, Raphael Ferreira, Alireza Zohourian, Rongxing Lu and A. Ghorbani. "CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment." Sensors (Basel, Switzerland), 23 (2023).

[9] Orkun Gurler. "COMPREHENSIVE ANALYSIS OF FUZZY C - MEANS CLUSTERING AND ITS VARIANTS." Journal of Modern Technology and Engineering (2024). https://doi.org/10.62476/jmte9269.

[10] Shan Hu. "Fuzzy C-means Clustering Algorithm and Its Application in Mental Health Intelligent Evaluation System." 2022 IEEE Conference on Telecommunications, Optics and Computer Science (TOCS) (2022): 1305-1309. https://doi.org/10.1109/tocs56154.2022.10016042.

[11] Fares, Hajar, Noura Aknin, Ghita Lazrek, and Mustapha Zeroual. "Intrusion Detection in IoT Environment Using Hyperparameters Tuned Machine and Deep Learning Models on the CICIoT2023 Dataset." Informatica 49, no. 5 (2025).

[12] Swati Lipsa, R. K. Dash and Nikola Ivković. "An interpretable dimensional reduction technique with an explainable model for detecting attacks in Internet of Medical Things devices." *Scientific Reports*, 15 (2025). https://doi.org/10.1038/s41598-025-93404-8.

[13] Tao Lei, Xiaohong Jia, Yanning Zhang, Lifeng He, Hongying Meng and A. Nandi. "Significantly Fast and Robust Fuzzy C-Means Clustering Algorithm Based on Morphological Reconstruction and Membership Filtering." *IEEE Transactions on Fuzzy Systems*, 26 (2018): 3027-3041. https://doi.org/10.1109/tfuzz.2018.2796074.

[14] Thiyagarajan, N., and N. Shanmugasundaram. "Accessing the Performance of K-Medoid, K-Means and FCM Clustering Techniques for Wireless Sensor Networks." In 2024 IEEE 5th India Council International Subsections Conference (INDISCON), pp. 1-5. IEEE, 2024.

[15] Mohammadrezapour, Omolbani, Ozgur Kisi, and Fariba Pourahmad. "Fuzzy c-means and K-means clustering with genetic algorithm for identification of homogeneous regions of groundwater quality." Neural Computing and Applications 32, no. 8 (2020): 3763-3775.

[16] Mohapatra, Niva, K. Shreya, and Ayes Chinmay. "Optimization of the random forest algorithm." In Advances in Data Science and Management".