NGUYỄN HẢI LÂM

HỆ THỐNG THÔNG TIN

2025

BỘ CÔNG THƯƠNG

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

NGUYỄN HẢI LÂM

NGHIÊN CỨU ỨNG DỤNG MÔ HÌNH MỜ VÀ LÝ THUYẾT ĐẠI SỐ GIA TỬ PHÁT HIỆN BẤT THƯỜNG TRONG MẠNG MÁY TÍNH

ĐỀ ÁN TỐT NGHIỆP THẠC SĨ HỆ THỐNG THÔNG TIN

Hà Nội – 2025

BỘ CÔNG THƯƠNG
ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

NGUYỄN HẢI LÂM

NGHIÊN CỨU ỨNG DỤNG MÔ HÌNH MỜ VÀ LÝ THUYẾT ĐẠI SỐ GIA TỬ PHÁT HIỆN BẤT THƯỜNG TRONG MẠNG MÁY TÍNH

Ngành: Hệ thống thông tin
Mã ngành: 8480104

ĐỀ ÁN TỐT NGHIỆP THẠC SĨ HỆ THỐNG THÔNG TIN

Người hướng dẫn thứ nhất: **TS. NGUYỄN VĂN THIỆN**

Người hướng dẫn thứ hai: **TS. HOÀNG VĂN THÔNG**

Hà Nội – 2025

LỜI CAM ĐOAN

Tôi là Nguyễn Hải Lâm, học viên cao học lớp Cao học Hệ thống thông tin, khóa 12. Tôi xin cam kết rằng đề tài thạc sĩ mang tên: "Nghiên cứu ứng dụng mô hình mờ và lý thuyết đại số gia tử phát hiện bất thường trong mạng máy tính" là kết quả nghiên cứu độc lập của tôi, được thực hiện dưới sự hướng dẫn khoa học của TS. Nguyễn Văn Thiện và TS. Hoàng Văn Thông.

Toàn bộ nội dung, kết quả nghiên cứu và dữ liệu trình bày trong đề tài này là trung thực, chưa từng công bố trong bất kỳ công trình nào khác. Tất cả số liệu dùng để phân tích, đánh giá đều được tôi thu thập từ các nguồn có trích dẫn rõ ràng trong phần Tài liệu tham khảo. Tôi khẳng định không vi phạm các nguyên tắc về đạo đức nghiên cứu khoa học trong quá trình thực hiện đề tài. Các tài liệu tham khảo được sử dụng chính xác, đúng mục đích và theo quy định học thuật.

Tôi hiểu rằng, nếu phát hiện hành vi gian lận nào, tôi sẽ chịu trách nhiệm hoàn toàn và có thể bị xem xét lại kết quả học tập, theo đúng quy định pháp luật và quy chế đào tạo hiện hành.

Hà Nội, ngày 01 tháng 07 năm 2025

Tác giả

LỜI CẢM ƠN

Để hoàn thành đề án thạc sĩ này, trước hết tôi xin bày tỏ lòng biết ơn sâu sắc tới TS. Nguyễn Văn Thiện và TS. Hoàng Văn Thông, những người đã tận tình hướng dẫn, định hướng và đồng hành cùng tôi trong suốt quá trình thực hiện đề tài. Qua sự chỉ bảo của hai thầy, tôi không chỉ lĩnh hội được phương pháp nghiên cứu khoa học mà còn tiếp thu nhiều kinh nghiệm quý báu về tư duy phản biện và tác phong làm việc chuyên nghiệp.

Tôi cũng xin chân thành cảm ơn quý thầy cô Trường Công nghệ Thông tin và Truyền thông – Trường Đại học Công nghiệp Hà Nội đã luôn hỗ trợ, tạo điều kiện thuận lợi cho tôi trong suốt thời gian học tập và nghiên cứu.

Đặc biệt, tôi xin gửi lời tri ân sâu sắc đến bố mẹ, những người luôn là điểm tựa tinh thần vững chắc, động viên và ủng hộ tôi không ngừng trong mọi hoàn cảnh.

Dù đã nỗ lực hết sức, đề án vẫn khó tránh khỏi những hạn chế do giới hạn thời gian và kinh nghiệm. Tôi rất mong nhận được ý kiến góp ý quý báu từ quý thầy cô và bạn bè để có thể hoàn thiện hơn trong các nghiên cứu sau này.

Tôi xin chân thành cảm ơn!

MỤC LỤC

DANH MỤC TỪ VIẾT TẮT

STT	Từ viết tắt	Ý nghĩa
1	AD	Phát hiện bất thường.
2	AE	Mạng mã hóa tự động – nén và tái tạo dữ liệu.
3	ANN	Mạng nơ-ron nhân tạo.
4	AS	Điểm bất thường.
5	AUC	Diện tích dưới đường cong ROC.
6	BL	Nhãn nhị phân.
7	CNN	Mạng nơ-ron tích chập – dùng trong ảnh.
8	DBN	Mạng niềm tin sâu – mô hình học sâu.
9	DDos	Tấn công từ chối dịch vụ phân tán.
10	DR	Tỷ lệ phát hiện.
11	DT	Cây quyết định.
12	FAR	Tỷ lệ báo động sai.
13	FRBS	Hệ thống suy diễn mờ dựa trên luật.
14	GAN	Mạng đối kháng sinh.
15	HA	Đại số gia tử
16	KDE	Ước lượng mật độ hạt nhân.
17	KNN	Thuật toán k láng giềng gần nhất.
18	LOF	Hệ số điểm ngoại lệ cục bộ.
19	NAD	Phát hiện bất thường mạng.
20	NIDS	Hệ thống phát hiện xâm nhập mạng.
21	OCC	Phân lớp một lớp.
22	OCSVM	Máy vector hỗ trợ một lớp.
23	OPHA	Phân hoạch tối ưu sử dụng đại số gia tử.
24	RNN	Mạng nơ-ron hồi tiếp – xử lý chuỗi.
25	ROC	Đường cong đặc trưng hoạt động.
26	SAE	Mạng mã hóa tự động có điều chuẩn.
27	SGERD	Phát hiện luật tiến hóa theo hạt ngữ nghĩa.
28	SVM	Máy vector hỗ trợ – phân loại dữ liệu.

DANH MỤC BẢNG BIỂU

DANH MỤC HÌNH ẢNH

MỞ ĐẦU

1. Lý do chọn đề tài

Khai phá dữ liệu và phát hiện tri thức được các nhà khoa học quan tâm nghiên cứu trong nhiều năm gần đây. Ứng dụng khai phá dữ liệu được thực hiện trong nhiều lĩnh vực khác nhau như giáo dục, y tế, an ninh, tài chính,… Đặc biệt, trong thời gian gần đây, khai phá dữ liệu và phát hiện tri thức trong lĩnh vực an toàn thông tin đang được quan tâm nghiên cứu.

Khi hạ tầng và dịch vụ mạng ngày càng mở rộng và hiện đại, các loại hình tấn công mạng cũng xuất hiện với tần suất ngày càng nhiều hơn và tinh vi hơn, gây ra nhiều mối nguy hiểm đặc biệt nghiêm trọng và tiềm ẩn trong hệ thống. Năm 2024 và 2025 chứng kiến sự leo thang đáng kể về các mối đe dọa an ninh mạng. Trí tuệ nhân tạo (AI) nổi lên như một yếu tố then chốt, vừa thúc đẩy khả năng tấn công tinh vi, vừa nâng cao năng lực phòng thủ. Trong năm 2024, ransomware tiếp tục thống trị, không chỉ mã hóa mà còn tống tiền bằng cách công khai dữ liệu đánh cắp. Các cuộc tấn công có chủ đích và tấn công chuỗi cung ứng gia tăng, khai thác lỗ hổng từ các nhà cung cấp bên thứ ba. Lừa đảo ngày càng tinh vi nhờ kỹ thuật xã hội. Lỗ hổng từ thiết bị biên và IoT cũng bị khai thác rộng rãi. Thiệt hại tài chính toàn cầu dự kiến đạt 9,5 nghìn tỷ USD. Dự báo 2025, AI sẽ định hình lại bối cảnh an ninh mạng. Tấn công do AI điều khiển sẽ bùng nổ, lợi dụng các mô hình LLM và Deepfake để lừa đảo và phát triển mã độc phức tạp. Ngược lại, AI cũng sẽ được tích hợp sâu hơn vào các giải pháp bảo mật để tự động hóa phát hiện và phản ứng.

Để đối phó, các tổ chức sẽ chú trọng kiến trúc Zero Trust, tăng cường bảo mật đám mây và dữ liệu nhạy cảm, đồng thời xây dựng khả năng phục hồi mạng. Trong thời đại số, việc nâng cao nhận thức về an toàn thông tin cùng với sự hợp tác giữa các quốc gia đóng vai trò then chốt trong việc bảo vệ tài sản kỹ thuật số. Việc nghiên cứu các phương pháp phát hiện và ngăn chặn các hành vi

tấn công mạng đã trở thành chủ đề thu hút sự chú ý của giới học thuật trong suốt nhiều thập kỷ. Một trong những hướng tiếp cận nổi bật là phát triển các hệ thống phát hiện xâm nhập mạng (Network Intrusion Detection Systems – NIDS).. NIDS được chia thành hai loại: phát hiện dựa trên dấu hiệu và phát hiện dựa trên sự bất thường [1]. Phân loại hệ thống phát hiện xâm nhập (NIDS) thường được thực hiện dựa trên phương pháp tiếp cận trong việc nhận diện hành vi xâm nhập. Các NIDS dựa trên chữ ký (Sigature-based) có khả năng phát hiện chính xác các cuộc tấn công đã được biết và định nghĩa từ trước. Tuy nhiên, chỉ các hệ thống dựa trên phát hiện bất thường (anomaly-based NIDS) mới có tiềm năng phát hiện các mối đe dọa chưa từng xuất hiện hoặc các hành vi tấn công mới chưa có mẫu chữ ký tương ứng. Trong lĩnh vực an toàn mạng, việc phát hiện các hành vi bất thường thường được biết đến với thuật ngữ "phát hiện bất thường mạng" (Network Anomaly Detection – NAD). Trên thực tế, một hệ thống phát hiện xâm nhập hiệu quả thường là sự kết hợp giữa nhiều thành phần, trong đó NAD đóng vai trò quan trọng ở trong hệ thống thông thường gồm có: phát hiện dựa trên chữ ký để xử lý các mối đe dọa đã biết và NAD nhằm xử lý các hành vi bất thường chưa xác định. Cốt lõi của nghiên cứu về NAD là xây dựng và tối ưu hóa mô hình bộ máy phát hiện bất thường, trong đó việc mô hình hóa hoạt động của hệ thống phát hiện đóng vai trò quan trọng trong việc nâng cao khả năng để phân biệt giữa mẫu dữ liệu bất thường so với bình thường, từ đó cải thiện hiệu suất phát hiện và giảm tỷ lệ cảnh báo sai.

Các phương pháp dành cho việc phát hiện bất thường theo hướng học máy thường tập trung vào việc đo lường độ lệch giữa dữ liệu đầu vào và mô hình hành vi bình thường đã được học từ trước. Mục tiêu là phát hiện các mẫu dữ liệu không phù hợp với hành vi hệ thống thông thường – được xem như là các dấu hiệu của xâm nhập hoặc tấn công mạng. Do đó, các giải pháp phát hiện bất thường yêu cầu hệ thống có khả năng học từ dữ liệu lịch sử nhằm xây dựng mô hình biểu diễn trạng thái hoạt động bình thường. Những kỹ thuật cho phép

hệ thống tự động học và trích xuất tri thức từ dữ liệu đầu vào để giải quyết các bài toán cụ thể được gọi chung là các phương pháp học máy (machine learning). Trong bối cảnh an toàn thông tin, phát hiện bất thường mạng là một trong những hướng nghiên cứu nhận được nhiều sự quan tâm, bởi nó cung cấp khả năng phát hiện các hình thức tấn công mới, chưa từng được ghi nhận trước đó – điều mà các hệ thống dựa trên chữ ký truyền thống không thể thực hiện được. Nhiều kỹ thuật học máy đã được các nhà nghiên cứu áp dụng thành công vào thực tế và mang lại kết quả đáng khích lệ trong lĩnh vực này như phương pháp dựa trên học sâu (Deep learning) [17, 36, 37], cây quyết định (Decision tree) [35], Máy véc tơ hỗ trợ (Support Vector Machine - SVM) [5], Hệ mờ dựa trên luật (Fuzzy Rule Based System - FRBS) [3,4,6,9,14-16,33].

Dựa trên nền tảng lý thuyết tập mờ và đại số gia tử, hệ thống suy diễn mờ dựa trên luật (Fuzzy Rule-Based Systems – FRBS) đã được phát triển như một công cụ tính toán mô phỏng hiệu quả quá trình suy luận và ra quyết định gần giống với tư duy con người. Với khả năng xử lý thông tin mơ hồ và không chắc chắn, FRBS đã chứng minh được tính ứng dụng rộng rãi trong nhiều lĩnh vực thực tiễn, đặc biệt là trong các bài toán phân loại như chẩn đoán y tế, đánh giá trong giáo dục, và các hệ thống hỗ trợ ra quyết định trong môi trường dữ liệu không chắc chắn. Ưu điểm của các phương pháp tiếp cận dựa trên FRBS là chi phí tính toán thấp, dễ hiểu với người sử dụng. Vấn đề phát hiện bất thường trong các mạng máy tính bản chất là bài toán phân loại trạng thái của hệ thống dựa trên các thông số của mạng để cảnh báo cho người quản trị thực hiện giám sát mạng. Với bài toán này chúng ta có thể phát triển các mô hình FRBS để phân loại trạng thái của mạng một cách hiệu quả theo thời gian thực. Do đó tôi đã chọn đề tài "*Nghiên cứu ứng dụng mô hình mờ và lý thuyết đại số gia tử phát hiện bất thường trong mạng máy tính*" làm đề tài cho đề án. Đề án sẽ tập trung nghiên cứu các phương pháp hiện bất thường theo các hướng tiếp cận khác nhau. Đi sâu nghiên cứu phương pháp xây dựng FRBS dựa trên lý thuyết

đại số gia tử và nghiên cứu ứng dụng vào phát hiện bất thương trong mạng máy tính.

2. Mục tiêu của đề tài

Mục tiêu của đề tài là xây dựng một mô hình phát hiện bất thường trong mạng máy tính dựa trên hệ thống suy diễn mờ (FRBS) kết hợp với đại số gia tử (HA), nhằm nâng cao độ chính xác và khả năng thích nghi của hệ thống khi xử lý các hành vi mạng bất thường. Đề tài hướng tới việc chứng minh tính hiệu quả của mô hình thông qua thực nghiệm trên tập dữ liệu thực tế, từ đó góp phần đề xuất một phương pháp khả thi ứng dụng trong các hệ thống giám sát an ninh mạng.

3. Nội dung nghiên cứu

- Nghiên cứu bài toán bất thường trong mạng máy tính

- Nghiên cứu các phương pháp tiếp cận khác nhau để giải quyết bài toán này: học máy có giám sát, học máy không giám sát, mạng mơ ron nhân tạo, học sâu.

- Nghiên cứu mô hình hệ mờ dạng luật sử dụng trong giải bài toán phân lớp.

- Lý thuyết tập mờ, lý thuyết đại số gia tử.

- Nghiên cứu phương pháp xây dựng FRBS dựa trên đại số gia tử

- Nghiên cứu ứng dụng FRBS được xây dựng dựa tren đại số gia tử trong phát hiện bất thường của mạng máy tính, nghiên cứu đánh giá phương pháp với các mô hình khác.

4. Phương pháp nghiên cứu

- Phương pháp nghiên cứu lý thuyết: Nghiên cứu về lý thuyết mờ, lý thuyết đại số gia tử, mô hình mờ, bài toán phát hiện bất thường nói chung và bài toán

phát hiện bất thường trong mạng máy tính nói riêng dựa trên các tài liệu đã được xuất bản và các tài liệu trên báo khoa học.

- Phương pháp thực nghiệm: Xây dựng mô hình mờ dựa trên lý thuyết mờ kết hợp với lý thuyết đại số gia tử từ dữ liệu thực tế và nghiên cứu đánh giá tính phù hợp của các mô hình đối với các mạng máy tính trong thực tế và so sánh đánh giá với các phương pháp tiếp khác.

5. Kết cấu đề án

Nội dung đề án tốt nghiệp gồm phần mở đầu, 3 chương và phần kết luận, cụ thể:

Chương 1 trình bày tổng quan kiến thức về phát hiện bất thường trong mạng máy tính. Các định nghĩa, các phương pháp phát hiện bất thường truyền thống, các phương pháp dựa trên học sâu.

Chương 2 trình bày tổng quan về lý thuyết tập mờ, lý thuyết đại số gia tử, hệ mờ dựa trên luật, các phương pháp lập luận dựa trên hệ mờ cho bài toán phân lớp, các tiêu chuẩn đánh giá hệ luật.

Chương 3 trình bày phương pháp xây dựng hệ luật mờ dựa trên đại số gia tử, mô tả tập dữ liệu sử dụng để xây dựng hệ luật phát hiện tấn công DDos, phân tích, đánh giá kết quả hệ luật thu được sau khi huấn luyện mô hình.

Cuối cùng đề án rút ra một số kết luận và hướng nghiên cứu phát triển tiếp theo của đề án.

CHƯƠNG 1. TỔNG QUAN VỀ PHÁT HIỆN BẤT THƯỜNG TRONG MẠNG MÁY TÍNH

1.1. PHÁT HIỆN BẤT THƯỜNG MẠNG MÁY TÍNH

Chương này giới thiệu tổng quan về đề tài, lý do chọn hướng nghiên cứu, mục tiêu và phạm vi triển khai, đồng thời xác định phương pháp tiếp cận và cấu trúc tổng thể của báo cáo. Trong bối cảnh các cuộc tấn công mạng ngày càng gia tăng về tần suất, mức độ tinh vi và khó nhận diện, nhu cầu phát triển các hệ thống phát hiện bất thường hiệu quả, linh hoạt và thích nghi tốt với môi trường dữ liệu phức tạp trở nên ngày càng cấp thiết.

Đề tài tập trung vào việc nghiên cứu và ứng dụng hệ thống suy diễn mờ dựa trên luật (FRBS) kết hợp với lý thuyết đại số gia tử (HA) trong phát hiện bất thường mạng. Phương pháp tiếp cận này hứa hẹn khắc phục được những hạn chế phổ biến của các hệ thống mờ truyền thống, như việc thiết lập hàm thành viên và sinh luật thường phụ thuộc chủ quan vào chuyên gia. Bằng cách tích hợp HA – một lý thuyết có khả năng định lượng và sắp xếp ngữ nghĩa các giá trị ngôn ngữ – vào FRBS, đề tài kỳ vọng xây dựng được mô hình phát hiện bất thường có độ chính xác cao hơn, dễ thích nghi với các loại dữ liệu thay đổi và khó xác định ranh giới rõ ràng.

Ngoài ra, chương 1 cũng làm rõ đối tượng, phạm vi nghiên cứu – cụ thể là các loại hành vi bất thường trong mạng máy tính, đặc biệt là tấn công từ chối dịch vụ phân tán (DDoS). Phương pháp nghiên cứu bao gồm cả phân tích lý thuyết và thực nghiệm, với dữ liệu thực tế được xử lý và đánh giá thông qua các chỉ số phổ biến như Precision, Recall, F1-score...

1.1.1. Định nghĩa

Trong [8] Hawkins định nghĩa "*Bất thường chỉ các mẫu dữ liệu được quan sát có sự sai lệch khá lớn so với các đối tượng quan sát khác như thể nó được tạo ra theo một cách thức hoàn toàn khác*".

Phát hiện bất thường (Anomaly Detection – AD) là một lĩnh vực nghiên cứu tập trung vào việc nhận diện những biểu hiện khác thường trong dữ liệu – tức những điểm dữ liệu thể hiện hành vi không phù hợp hoặc lệch đáng kể so với xu hướng chung của toàn bộ tập quan sát. Các đối tượng được coi là bất thường thường phản ánh các tình huống không phổ biến, hiếm gặp hoặc tiềm ẩn rủi ro, chẳng hạn như gian lận tài chính, lỗi thiết bị, hay dấu hiệu của một cuộc tấn công mạng. Việc phát hiện chính xác những bất thường này đóng vai trò quan trọng trong công tác giám sát hệ thống, cảnh báo sớm và bảo vệ dữ liệu trong nhiều lĩnh vực.

Sự quan tâm đến AD ngày càng gia tăng, không chỉ trong giới học thuật mà còn trong thực tiễn công nghiệp, bởi khả năng ứng dụng rộng rãi của nó trong nhiều ngành nghề như chăm sóc sức khỏe (phát hiện bệnh lý), tài chính (nhận diện gian lận), công nghiệp sản xuất (kiểm tra lỗi sản phẩm), và đặc biệt là trong lĩnh vực an toàn thông tin. Để giải quyết bài toán phát hiện bất thường, nhiều phương pháp tiếp cận đã được đề xuất, trong đó phổ biến nhất là các kỹ thuật thống kê, học máy, và logic mờ. Những phương pháp này không ngừng được cải tiến nhằm nâng cao khả năng phát hiện chính xác cũng như tăng cường khả năng tổng quát hóa mô hình đối với các tình huống chưa từng gặp trước đó.

Ngoài tên gọi “phát hiện bất thường”, lĩnh vực này còn được biết đến với các thuật ngữ đồng nghĩa như phát hiện mẫu mới (novelty detection) hay phát hiện sai lệch (deviation detection). Trong bối cảnh an ninh mạng, khái niệm AD được cụ thể hóa dưới dạng phát hiện bất thường mạng (Network Anomaly

Detection – NAD), nhằm nhận diện những hành vi hoặc lưu lượng truyền thông không tương thích với đặc điểm thông thường của hệ thống mạng đang giám sát. Các dấu hiệu bất thường có thể phát sinh từ nhiều nguyên nhân khác nhau như tấn công mạng có chủ đích, cấu hình hệ thống không đúng, hoặc hành vi người dùng không tuân thủ chính sách bảo mật. Để hỗ trợ phân tích hiệu quả, NAD thường chia các loại bất thường thành ba nhóm cơ bản, dựa trên mức độ khác biệt và bản chất của hành vi phát hiện được trong dòng dữ liệu mạng.Trong phát hiện bất thường, có thể phân biệt hai dạng phổ biến dựa trên cách thức biểu hiện của hành vi bất thường: (1) **bất thường theo tập hợp mẫu** – tức là một hành vi chỉ được nhận diện là bất thường khi xuất hiện đồng thời trong một nhóm các mẫu dữ liệu; và (2) **bất thường theo ngữ cảnh** – khi một điểm dữ liệu chỉ trở nên bất thường trong một ngữ cảnh hoặc điều kiện cụ thể.

Trong phạm vi đề án này, các hành vi bất thường được định nghĩa là các dạng tấn công mạng hoặc hành vi gây rối làm sai lệch chức năng thông thường của hệ thống. Cụ thể:

U2R (User to Root) là dạng tấn công leo thang đặc quyền, trong đó kẻ tấn công cố gắng chiếm quyền truy cập root từ một tài khoản người dùng bình thường;

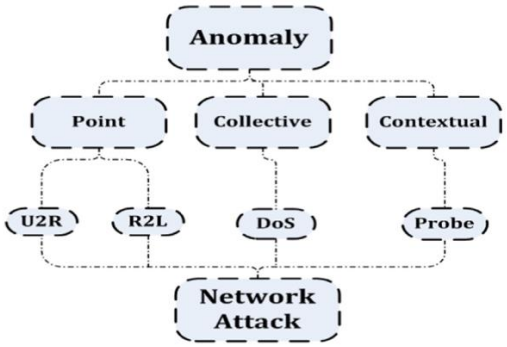
R2L (Remote to Local) là dạng tấn công trong đó kẻ tấn công từ xa cố gắng giành quyền điều khiển hệ thống cục bộ;

DoS (Denial of Service) là các tấn công từ chối dịch vụ, trong đó một yêu cầu riêng lẻ có thể là hợp lệ, nhưng khi xuất hiện với số lượng lớn trong một **tập hợp mẫu** sẽ gây quá tải và làm gián đoạn dịch vụ – đây là biểu hiện điển hình của bất thường theo nhóm;

Probe là dạng tấn công dò quét hệ thống để thu thập thông tin, thường bao gồm các gói tin hỏi/đáp gửi đến hệ thống trong khi các thông số hạ tầng và dịch

vụ mạng vẫn ổn định – do đó hành vi này chỉ được xem là bất thường trong **ngữ cảnh** cụ thể.

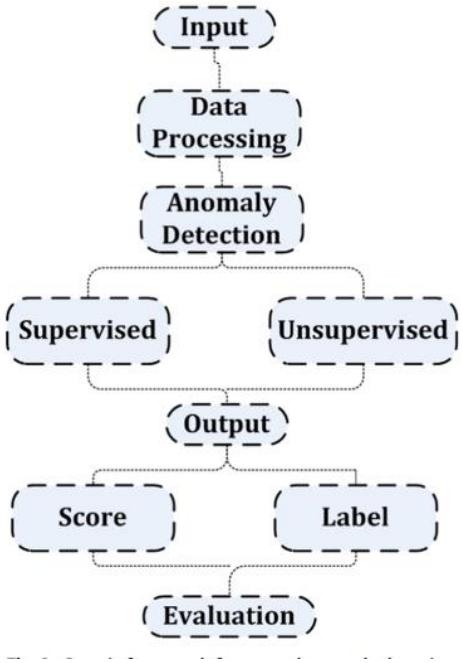
Trong các dạng bất thường mạng, đề tài có đề cập tới đầy đủ các loại hình phổ biến như U2R, R2L, DoS, Probe. Tuy nhiên, trong phạm vi thực nghiệm, đề tài lựa chọn tấn công **DDoS** để làm trường hợp điển hình nhằm kiểm thử và đánh giá hiệu quả của mô hình phát hiện bất thường dựa trên FRBS và đại số gia tử. Đây là bước tiền đề để từ đó có thể mở rộng ứng dụng mô hình cho các loại bất thường khác trong tương lai.



Hình 1.1: Các nhóm phân loại tấn công mạng và loại bất thường [27]

1.1.2. Các phương pháp phát hiện bất thường mạng máy tính

Cấu trúc tổng quan của hệ thống phát hiện bất thường mạng được minh họa trong Hình 1.2. Theo đó, luồng dữ liệu mạng thô sau khi thu thập sẽ trải qua các bước trích chọn đặc trưng và tiền xử lý (như làm sạch, chuẩn hóa, và biến đổi dữ liệu). Kết quả tiền xử lý này sau đó được đưa vào mô-đun phát hiện bất thường—một mô hình phân lớp NAD—đóng vai trò cốt lõi trong việc phân biệt giữa lưu lượng bình thường và các kiểu tấn công tiềm ẩn.



Hình 1.2: Khung kiến trúc ác mô hình phát triển bất thường mạng [27]

Các phương pháp xây dựng mô hình nhằm giải quyết bài toán phát hiện bất thường mạng (NAD) đã được nghiên cứu và phát triển theo nhiều hướng tiếp cận khác nhau. Trong đó, hướng ứng dụng học máy (machine learning) nổi lên như một trong những giải pháp được chú trọng và khai thác rộng rãi nhờ vào khả năng học tự động và thích nghi với dữ liệu lớn, đa dạng. Các thuật toán học máy có thể được phân loại thành ba nhóm chính: học có giám sát (supervised learning), học không giám sát (unsupervised learning) và học bán giám sát (semi-supervised learning)—mỗi nhóm có đặc trưng và điều kiện áp dụng riêng biệt.

- Học có giám sát yêu cầu tập dữ liệu huấn luyện phải được gán nhãn rõ ràng cho từng mẫu, phân biệt giữa hành vi "bình thường" và "bất thường". Dựa vào thông tin nhãn này, mô hình học có thể tối ưu hóa quá trình phân loại với độ chính xác cao. Các thuật toán tiêu biểu trong nhóm này bao gồm: mạng nơ-ron nhân tạo (ANN), máy vector hỗ trợ (SVM), K-nearest neighbors (KNN), mạng Bayes, cây quyết định (DT) và hệ mờ suy diễn theo luật (FRBS

- Ngược lại, học không giám sát hoạt động dựa trên giả định rằng phần lớn lưu lượng mạng là bình thường, còn các mẫu bất thường chỉ chiếm tỷ lệ rất nhỏ. Các mô hình thuộc nhóm này không cần đến thông tin nhãn trong quá trình huấn luyện, mà tự động học cấu trúc, đặc điểm phân bố của dữ liệu bình thường, từ đó nhận diện những điểm lệch (outliers) hoặc cụm dữ liệu có mật độ thấp.
- Học bán giám sát kết hợp ưu điểm của cả hai hướng trên bằng cách sử dụng một tập nhỏ đã gán nhãn (thường chỉ gồm mẫu "bình thường") và một tập lớn không nhãn. Cách tiếp cận này giảm thiểu gánh nặng gán nhãn nhưng vẫn khai thác được thông tin nhãn để cải thiện độ chính xác. Trong an ninh mạng, các thuật toán như One-Class SVM (OCSVM) [5], Local Outlier Factor (LOF) [20] hay Kernel Density Estimation (KDE) [22] thường được áp dụng theo mô hình bán giám sát, khi chỉ dữ liệu bình thường được dùng để học cấu trúc, rồi bất thường được phát hiện dựa trên độ lệch so với mô hình đó.

1.1.3. Lưu lượng mạng máy tính

Trong các hệ thống phát hiện bất thường trên mạng máy tính, nguồn dữ liệu đầu vào chủ yếu là lưu lượng mạng (network traffic) được thu thập bằng các công cụ chuyên dụng như trình bắt gói tin (sniffer). Dữ liệu này thường bao gồm các gói tin đã được xử lý và định dạng theo cấu trúc chuẩn, cho phép hệ thống phân tích sâu hơn ở các cấp độ khác nhau. Một số hệ thống phát hiện xâm nhập điển hình, chẳng hạn như Snort [21], thực hiện phát hiện bất thường trực tiếp trên các gói tin thô mà không cần tái cấu trúc lưu lượng. Tuy nhiên, trong nhiều trường hợp, các cuộc tấn công mạng chỉ có thể được phát hiện rõ ràng khi phân tích theo cấp độ phiên (session) hoặc luồng dữ liệu (flow). Điều này đòi hỏi quá trình tiền xử lý dữ liệu (preprocessing) nhằm chuyển đổi dữ liệu thô thành dạng thích hợp để trích xuất đặc trưng ở cả hai cấp độ: gói và phiên.

Trích chọn đặc trưng (feature extraction) đóng vai trò then chốt trong việc mô hình hóa các khía cạnh quan trọng của lưu lượng mạng, giúp nâng cao hiệu quả trong việc phát hiện hành vi bất thường. Quá trình này có thể sử dụng nhiều kỹ thuật khác nhau, từ các phương pháp thống kê đơn giản đến các thuật toán học máy hiện đại. Các đặc trưng được trích chọn từ lưu lượng thường được phân chia thành hai nhóm chính theo kiểu dữ liệu: đặc trưng định lượng (numerical) và đặc trưng phân loại (categorical). Trong đó, đặc trưng định lượng lại được chia thành.

- Dạng rời rạc (discrete) – thể hiện các thuộc tính có thể đếm được như số lượng gói tin hoặc số lần kết nối

- Dạng liên tục (continuous) – mô tả các giá trị đo lường có thể biến thiên, như thời gian kết nối hoặc tỷ lệ truyền tải

Việc lựa chọn các đặc trưng phù hợp là bước quan trọng trong quá trình xây dựng mô hình, không chỉ giúp giảm chiều dữ liệu (dimensionality reduction) mà còn tăng hiệu suất xử lý, đồng thời cải thiện khả năng giải thích kết quả và phát hiện chính xác hơn. Dựa trên cấu trúc lưu lượng mạng, các đặc trưng thường được phân loại thành ba nhóm chính:

- Đặc trưng cơ bản (Basic Features): Bao gồm các thông tin có thể trích xuất trực tiếp từ phiên giao tiếp TCP/IP, chẳng hạn như địa chỉ IP nguồn và đích, cổng giao tiếp, thời lượng phiên, số lượng gói gửi và nhận. Đây là những đặc trưng đơn giản nhưng quan trọng, giúp định hình hoạt động cơ bản của một kết nối.

- Đặc trưng lưu lượng (Traffic Features): Được suy luận từ các trường trong tiêu đề của gói tin (TCP/IP header), như kích thước cửa sổ (window size), thể hiện đặc điểm hành vi truyền thông trong quá trình trao đổi dữ liệu giữa hai đầu phiên kết nối.

- Đặc trưng nội dung (Content Features): Được lấy từ phần dữ liệu (payload) của gói tin, nhóm đặc trưng này cung cấp thông tin chuyên sâu hơn về nội dung truyền tải, và thường được khai thác để nhận diện dấu hiệu tấn công, mã độc hoặc các hành vi xâm nhập ẩn trong nội dung phiên.

Trong các hệ thống phát hiện xâm nhập mạng (Network Intrusion Detection Systems – NIDS), dữ liệu đầu vào thường được xây dựng từ các nguồn lưu lượng mạng thực tế hoặc mô phỏng, sau đó áp dụng kỹ thuật trích chọn đặc trưng phù hợp với mục tiêu phân tích. Mục tiêu cuối cùng là đánh giá mức độ hiệu quả và độ tin cậy của các thuật toán bảo mật trong việc phân biệt hành vi bình thường và các dấu hiệu bất thường, nhằm kịp thời ngăn chặn các mối đe dọa mạng.

1.1.4. Đầu ra của mô hình xác định bất thường

Trong các mô hình phát hiện bất thường, thông tin đầu ra thường được thể hiện dưới hai hình thức phổ biến: điểm số bất thường (Anomaly Score – AS) và nhãn nhị phân (Binary Label – BL). Điểm số bất thường là một giá trị liên tục phản ánh mức độ lệch chuẩn của một mẫu dữ liệu so với hành vi thông thường trong tập huấn luyện. Trong khi đó, nhãn nhị phân phân loại trực tiếp từng mẫu là "bình thường" hoặc "bất thường".

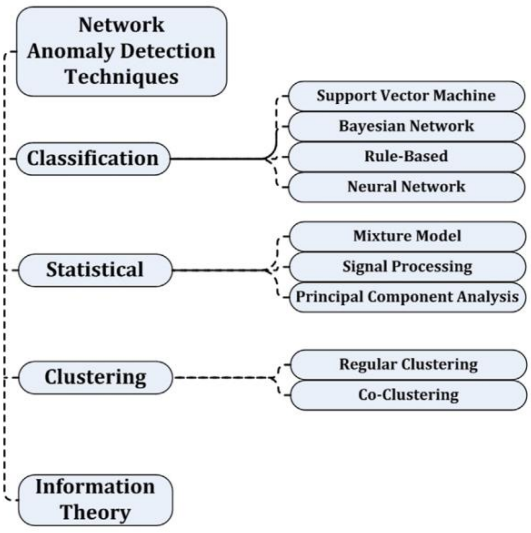
Trên thực tế, nhiều hệ thống ứng dụng ưu tiên lựa chọn đầu ra theo dạng nhị phân do tính đơn giản và dễ triển khai. Ngược lại, đầu ra dạng điểm số thường yêu cầu thêm giai đoạn hậu xử lý (post-processing) để xác định ngưỡng phân loại phù hợp. Việc thiết lập ngưỡng này có thể phụ thuộc vào đặc thù bài toán, yêu cầu vận hành hoặc kinh nghiệm của chuyên gia, từ đó làm gia tăng tính phức tạp và không đảm bảo tính đồng nhất trong các hệ thống thực tế. Tuy nhiên, nếu được điều chỉnh tốt, điểm số bất thường lại có thể cung cấp thông tin chi tiết hơn về mức độ nguy cơ của một sự kiện, hỗ trợ đánh giá ưu tiên trong quá trình xử lý cảnh báo.

- Độ đo bất thường (AS): Mô hình sẽ gán cho mỗi điểm dữ liệu một giá trị xác suất thuộc khoảng (0,1), phản ánh mức độ bất thường tương đối. Tuy nhiên, để chuyển giá trị này thành quyết định rõ ràng về trạng thái bình thường hay bất thường, hệ thống cần xác định một ngưỡng (threshold) thích hợp. Việc lựa chọn ngưỡng không tối ưu có thể ảnh hưởng tiêu cực đến hiệu quả phát hiện.

- Nhãn nhị phân (BL): Với dạng đầu ra này, mô hình đánh dấu điểm dữ liệu là "1" nếu được xem là bất thường và "0" nếu bình thường. Phương pháp này hỗ trợ trực tiếp cho các hệ thống giám sát theo thời gian thực, nhờ khả năng đưa ra quyết định rõ ràng. Ngoài ra, các mô hình BL cũng được đánh giá là phù hợp hơn trong bài toán phát hiện dị thường chưa từng gặp trước đó và cung cấp thông tin rõ ràng hơn so với mô hình dựa trên điểm số bất thường.

Tóm lại, trong bối cảnh triển khai thực tế, hệ thống phát hiện bất thường thường yêu cầu đầu ra mang tính quyết định cao, thay vì chỉ cung cấp chỉ số đánh giá mức độ dị thường.

1.2. MỘT SỐ PHƯƠNG PHÁP PHÁT HIỆN BẤT THƯỜNG MẠNG



Hình 1.3: Phân loại các kỹ thuật xây dựng mô hình NAD [27]

Các phương pháp phát hiện bất thường hiện nay chủ yếu tận dụng các kỹ thuật từ thống kê truyền thống, khai phá dữ liệu (data mining) và đặc biệt là học máy (machine learning) để xây dựng các mô hình hiệu quả. Tuy nhiên, việc phân loại hệ thống các thuật toán AD (Anomaly Detection) vẫn chưa đạt được sự thống nhất rõ ràng trong cộng đồng nghiên cứu, bởi tồn tại nhiều góc nhìn khác nhau cũng như sự giao thoa về mặt phương pháp giữa các trường phái tiếp cận. Trong số các hướng tiếp cận đang được phát triển, hình 1.2 đưa ra một cách nhìn tổng quát về các nhóm kỹ thuật thường được ứng dụng trong phát hiện bất thường mạng (Network Anomaly Detection – NAD).

Đáng chú ý trong các kỹ thuật học máy là nhóm phân lớp đơn (one-class classification), vốn được xem là rất phù hợp đối với các ứng dụng trong lĩnh vực an ninh mạng. Lý do là vì các hệ thống bảo mật thực tế thường chỉ có sẵn dữ liệu về hành vi bình thường, trong khi dữ liệu bất thường (tấn công, bất hợp lệ) lại hiếm, không đầy đủ hoặc hoàn toàn chưa từng xuất hiện

Phân lớp đơn cho phép mô hình học được ranh giới mô tả lớp "bình thường" và phát hiện bất kỳ điểm dữ liệu nào nằm ngoài ranh giới đó là bất thường. Ngoài khả năng hoạt động tốt trong không gian đặc trưng có số chiều cao, các thuật toán phân lớp đơn còn hỗ trợ hiệu quả quá trình ước lượng siêu tham số, từ đó nâng cao độ chính xác trong việc nhận diện các mẫu không quen thuộc. Điều này đặc biệt hữu ích khi đối mặt với các cuộc tấn công dạng mới hoặc phần mềm độc hại chưa từng được ghi nhận trước đó, Các kỹ thuật phân lớp đơn có thể chia thành hai nhóm chính dựa trên bản chất thuật toán:

- Phân lớp đơn truyền thống: Bao gồm những thuật toán đã được kiểm chứng như One-Class SVM, K-Nearest Neighbors (k-NN), hoặc các phương pháp dựa trên mật độ như Gaussian hoặc Parzen Window. Những phương pháp này thường đơn giản, dễ triển khai và phù hợp với bài toán có dữ liệu giới hạn.

- Phân lớp đơn dựa trên học sâu: Nhóm này ứng dụng các mô hình mạng nơ-ron sâu (deep neural networks) như Autoencoder hoặc Generative Adversarial Networks (GANs), nhằm học biểu diễn tiềm ẩn của dữ liệu bình thường, từ đó phát hiện sự khác biệt ở những quan sát bất thường chưa từng thấy.

Trong kiến trúc tổng thể của các hệ thống NAD, các thuật toán phân lớp đơn có thể được sử dụng như thành phần lõi để trực tiếp xử lý dữ liệu đặc trưng đầu vào. Ngoài ra, chúng cũng thường được tích hợp sau các bước giảm số chiều đặc trưng (dimensionality reduction), đóng vai trò như một mô-đun phân tích tinh, giúp tăng cường khả năng tách biệt giữa các mẫu dữ liệu bất thường và bình thường một cách tối ưu hơn.

1.2.1. Một số phương pháp phân lớp đơn truyền thống

Các thuật toán phân lớp đơn truyền thống đã cho thấy hiệu quả rõ rệt trong việc nhận diện các hành vi bất thường trong môi trường mạng. Nhờ khả năng xử lý tốt dữ liệu có đặc điểm phức tạp và không gian đặc trưng nhiều chiều, một số phương pháp kinh điển dưới đây đã được ứng dụng phổ biến trong các hệ thống phát hiện xâm nhập mạng:

Local Outlier Factor (LOF) [20]: Đây là một kỹ thuật đặc biệt hiệu quả đối với các tập dữ liệu có chiều không gian cao. LOF đánh giá mức độ bất thường của một điểm dựa trên mật độ cục bộ của khu vực lân cận, bằng cách so sánh mật độ của điểm đang xét với mật độ của các điểm xung quanh nó. Những điểm có mật độ thấp hơn đáng kể so với hàng xóm sẽ được gán là bất thường

Kernel Density Estimation (KDE) [22]: KDE là một phương pháp học không giám sát hoạt động bằng cách ước lượng hàm mật độ xác suất của dữ liệu mà không cần giả định cụ thể về hình dạng phân phối nền. Nhờ vào tính

linh hoạt trong việc mô hình hóa phân bố, KDE đặc biệt phù hợp với các tập dữ liệu mạng có cấu trúc phân tán phức tạp hoặc không tuyến tính

One-Class Support Vector Machine (OCSVM) [5]: Là một biến thể của SVM, OCSVM được thiết kế riêng cho bài toán phát hiện bất thường bằng cách xây dựng một siêu mặt phân tách sao cho phần lớn dữ liệu "bình thường" nằm trong biên an toàn, trong khi các điểm nằm ngoài vùng này được xem là bất thường. Nhờ khả năng tổng quát hóa tốt, OCSVM có thể thích nghi với nhiều loại dữ liệu khác nhau và được sử dụng trong nhiều lĩnh vực ứng dụng

Nhìn một cách tổng thể, các phương pháp phân lớp đơn truyền thống có thể được phân loại dựa theo nguyên lý cốt lõi của chúng thành hai nhóm chính: phương pháp dựa trên khoảng cách (distance-based methods) và phương pháp dựa trên mật độ (density-based methods) [34].

- Các phương pháp dựa trên khoảng cách đánh giá sự bất thường thông qua việc đo lường độ lệch giữa một điểm dữ liệu với các điểm lân cận—khoảng cách càng lớn, khả năng là điểm bất thường càng cao
- Ngược lại, các phương pháp dựa trên mật độ tập trung vào việc phát hiện các điểm nằm trong vùng có mật độ thấp hơn đáng kể so với vùng xung quanh, vì đây là đặc điểm phổ biến của các bất thường trong phân bố dữ liệu thực tế

a. Phương pháp phân lớp đơn dựa trên khoảng cách

Các phương pháp phát hiện bất thường dựa trên khoảng cách (distance-based anomaly detection) khai thác ý tưởng rằng các điểm dữ liệu bất thường thường có vị trí cách biệt rõ rệt so với phần lớn các điểm còn lại trong không gian đặc trưng. Để đo lường mức độ lệch này, các thuật toán thường sử dụng các hàm đo khoảng cách như khoảng cách Euclid hoặc các hàm tương tự nhằm định lượng sự khác biệt giữa các quan sát. Ý tưởng trung tâm của hướng tiếp

cận này là: nếu một điểm dữ liệu nằm ở khoảng cách lớn so với các cụm dữ liệu bình thường, thì điểm đó có khả năng cao là bất thường.

Một kỹ thuật điển hình trong nhóm này là phương pháp láng giềng gần nhất (Nearest Neighbor – NN), trong đó mức độ bất thường của một quan sát được xác định bằng khoảng cách đến các điểm dữ liệu lân cận gần nhất. Phương pháp này dựa trên giả định rằng dữ liệu bình thường thường phân bố dày đặc và gần nhau, trong khi các điểm bất thường sẽ xuất hiện rời rạc và nằm xa vùng mật độ cao. Do đó, một điểm càng xa các "láng giềng" của nó thì càng có xác suất là bất thường.

Các thuật toán trong nhóm này phụ thuộc vào hai yếu tố chính:

Định nghĩa về quan hệ láng giềng (neighbor definition): Có thể thiết lập theo nhiều cách, phổ biến nhất là dựa trên khoảng cách trực tiếp giữa các điểm dữ liệu (distance-based kernel) – như trong K-Nearest Neighbors (KNN), hoặc dựa vào mật độ lân cận (local density-based kernel) – như thuật toán Local Outlier Factor (LOF). Trong trường hợp thứ hai, thuật toán không chỉ xét đến khoảng cách mà còn phân tích mức độ mật độ tương đối giữa điểm đang xét và khu vực xung quanh nó, giúp tăng khả năng phát hiện bất thường trong các vùng phân bố không đồng đều.

Số lượng K láng giềng gần nhất: Tham số này ảnh hưởng trực tiếp đến độ chính xác và độ nhạy của thuật toán.

Tuy nhiên, các phương pháp dựa trên khoảng cách thường gặp khó khăn khi áp dụng với dữ liệu có kích thước lớn hoặc không gian đặc trưng nhiều chiều (high-dimensional data), do chi phí tính toán khoảng cách tăng theo cấp số mũ và hiện tượng "curse of dimensionality". Việc xác định giá trị K tối ưu cũng là một thách thức chưa có lời giải phổ quát.

Trong số các phương pháp phát hiện bất thường mạng, Local Outlier Factor (LOF) [20] được xem là một thuật toán nổi bật nhờ khả năng kết hợp

giữa hai yếu tố quan trọng: khoảng cách hình học và mật độ cục bộ của dữ liệu. Mặc dù thường được xếp vào nhóm kỹ thuật dựa trên mật độ, LOF vẫn sử dụng cơ chế tính k-láng giềng gần nhất (k-nearest neighbors) như một phần cốt lõi trong việc đánh giá mức độ bất thường của từng điểm dữ liệu.

Thuật toán LOF vận hành theo một chuỗi các bước tính toán như sau:

1. Khởi tạo và xác định lân cận:

Giả sử có một tập dữ liệu huấn luyện $P = \{p1, p2, ..., pn\} \subset \mathbb{R}^d$, trong đó dist(p, q) là khoảng cách giữa hai điểm bất kỳ $p, q \in P$. Gọi $D_k(p)$ là khoảng cách từ điểm p đến láng giềng thứ k, còn $L_k(p)$ là tập gồm k điểm gần nhất với p.

2. Xác định khả năng tiếp cận (reachability distance):

Với mỗi cặp (p, q), trong đó $q \in L_k(p)$, khoảng cách tiếp cận được định nghĩa là:

$$R_k(p, q) = \max(\text{dist}(p, q), D_k(p)) \qquad (1.1)$$

3. Tính trung bình khả năng tiếp cận (average reachability distance):

Mức độ tiếp cận trung bình của điểm p, ký hiệu là $AR_k(p)$, được tính như sau:

$$AR_k(p) = (1 / |L_k(p)|) \times \sum R_k(p, o) \text{ với } o \in L_k(p) \qquad (1.2)$$

4. Tính chỉ số LOF:

Hệ số bất thường cục bộ của một điểm p, tức Local Outlier Factor, phản ánh mức độ biệt lập của điểm đó so với các láng giềng, được tính bằng tỷ lệ giữa giá trị trung bình khả năng tiếp cận của các láng giềng và chính điểm đó:

$$LOF_k(p) = (\sum AR_k(o)) / (|L_k(p)| \times AR_k(p)) \text{ với } o \in L_k(p) \qquad (1.3)$$

Nếu chỉ số LOF của một điểm lớn hơn đáng kể so với ngưỡng cho trước, điểm đó có thể được xem là bất thường. Ngược lại, các điểm có giá trị LOF gần 1 thường là bình thường.

Thuật toán LOF cho phép đánh giá bất thường một cách linh hoạt mà không cần giả định về phân bố dữ liệu. Tuy nhiên, hiệu quả của phương pháp này phụ thuộc nhiều vào ngưỡng phân loại được chọn thủ công – một bước vẫn cần đến sự điều chỉnh từ chuyên gia nhằm đảm bảo cân bằng giữa phát hiện đúng và tránh báo động giả.

Mặc dù LOF có khả năng phát hiện hiệu quả các điểm bất thường trong các vùng dữ liệu có mật độ thay đổi, nhược điểm chính của nó nằm ở chi phí tính toán cao, đặc biệt là trong các tập dữ liệu lớn với số chiều cao. Cụ thể, việc tính toán giá trị $R_k(p, q)$ và danh sách láng giềng đòi hỏi nhiều phép đo khoảng cách, ảnh hưởng đến hiệu suất toàn hệ thống. Trong thực tế, LOF thường được kết hợp với các kỹ thuật giảm chiều, tiền xử lý hoặc học sâu để cải thiện hiệu quả vận hành trong các hệ thống NAD hiện đại [38].

Dẫu vậy, LOF vẫn gặp khó khăn khi áp dụng cho dữ liệu có tính chất phân tán và chiều cao (sparse and high-dimensional), bên cạnh đó, việc xác định ngưỡng vẫn không thể hoàn toàn tự động hóa mà cần sự hỗ trợ từ chuyên gia.

b. Phương pháp phân lớp đơn dựa trên mật độ

Các phương pháp phát hiện bất thường dựa trên mật độ (density-based anomaly detection) vận hành theo nguyên lý rằng dữ liệu "bình thường" thường có xu hướng tập trung tại các vùng có xác suất xuất hiện cao, trong khi các điểm bất thường thường xuất hiện ở những khu vực hiếm gặp, có xác suất thấp. Do đó, bằng cách ước lượng hàm mật độ xác suất của phân phối dữ liệu huấn luyện và thiết lập một ngưỡng loại trừ (threshold), hệ thống có thể phân biệt các điểm có khả năng là bất thường nếu giá trị xác suất tương ứng của chúng nằm dưới ngưỡng đã chọn. Một điểm đáng chú ý là trong quá trình huấn luyện,

chỉ dữ liệu bình thường được sử dụng để xây dựng mô hình học, nhằm tạo nên một biểu diễn đáng tin cậy cho hành vi hợp lệ.

Một trong những thách thức lớn nhất của hướng tiếp cận này nằm ở việc ước lượng chính xác hàm mật độ xác suất của dữ liệu bình thường – đặc biệt khi không biết trước dạng phân phối. Có hai kỹ thuật nổi bật thường được sử dụng để giải quyết bài toán này: Mô hình Hỗn hợp Gauss (Gaussian Mixture Models – GMMs) và Phép xấp xỉ Mật độ Nhân (Kernel Density Estimation – KDE).

Trong nghiên cứu của M.P. Wand và cộng sự [22], KDE được trình bày như một phương pháp ước lượng mật độ mềm dẻo, sử dụng các hàm nhân (kernel functions) để ước lượng mật độ tại từng điểm quan sát. Khác với các phương pháp tham số truyền thống, KDE cho phép mô hình phân bố của dữ liệu theo cách tự nhiên và linh hoạt hơn, phù hợp với các tập dữ liệu có đặc điểm phức tạp hoặc phân bố phi tuyến.

Với khả năng thích ứng cao và không yêu cầu cấu trúc phân phối cố định, KDE đã chứng tỏ tính hiệu quả trong các bài toán phát hiện bất thường dạng phân lớp đơn (one-class classification). Đặc biệt, trong các hệ thống an ninh mạng hiện đại, KDE thường được sử dụng như một thành phần quan trọng để xây dựng mô hình nhận diện hành vi bất thường, nhờ vào khả năng mô phỏng chính xác các vùng mật độ cao cũng như phát hiện các mẫu lệch chuẩn nằm ngoài biên phân phối dữ liệu huấn luyện.

Giả sử có một tập dữ liệu $y = \{y_1, y_2, ..., y_n\} \in \mathbb{R}^d$ được lấy mẫu từ một phân phối xác suất chưa biết trước, với hàm mật độ xác suất tương ứng là $p(y)$. Mục tiêu của phương pháp Kernel Density Estimation (KDE) là xây dựng một hàm ước lượng mật độ xác suất $\hat{p}(y)$ tại điểm y, được xác định thông qua công thức:

$$\hat{p}(y) = (1/n) \sum K_h(y - y_i) \qquad (1.4)$$

trong đó K_h là một hàm nhân (kernel function) với tham số điều chỉnh h gọi là băng thông (bandwidth). Hàm K_h: $\mathbb{R} \rightarrow \mathbb{R}$ có vai trò xác định mức ảnh hưởng của từng điểm dữ liệu y_i đối với điểm đang xét y.

Hiệu quả của KDE phụ thuộc đáng kể vào hai yếu tố chính: loại hàm nhân được lựa chọn và giá trị của tham số băng thông h. Trong thực tế, mỗi điểm dữ liệu trong tập huấn luyện đóng góp một thành phần vào tổng mật độ thông qua hàm nhân đặt tại chính điểm đó. Một số hàm nhân phổ biến bao gồm Gaussian, Uniform và Exponential. Trong nghiên cứu này, hàm nhân Gaussian được sử dụng do khả năng tạo ra phân bố mượt và ổn định, được định nghĩa như sau:

$$K_h(y) = \exp(-y^2 / (2h^2)) \qquad (1.5)$$

Tham số băng thông h kiểm soát sự đánh đổi giữa độ lệch (bias) và phương sai (variance) của mô hình. Khi h lớn, hàm mật độ trở nên mượt mà hơn, giúp giảm phương sai nhưng có thể gây ra độ lệch lớn. Ngược lại, nếu h quá nhỏ, phân bố trở nên nhọn và nhạy hơn với biến động dữ liệu, làm tăng phương sai nhưng giảm độ lệch. Việc lựa chọn giá trị h tối ưu là yếu tố then chốt để mô hình KDE đạt hiệu năng cao nhất.

Trong những năm gần đây, KDE đã được ứng dụng rộng rãi như một thành phần cốt lõi trong các hệ thống phát hiện bất thường mạng (NAD). Khi được huấn luyện với dữ liệu hoàn toàn bình thường, mô hình có thể ước lượng mật độ xác suất cho các điểm mới trong giai đoạn kiểm thử. Nếu xác suất ước lượng của một điểm thấp hơn ngưỡng xác định trước, điểm đó sẽ được gán nhãn là bất thường.

Tuy nhiên, cũng giống như nhiều mô hình phân lớp đơn khác, việc xác định ngưỡng quyết định phù hợp trong thực tế không hề đơn giản. Ngưỡng này ảnh hưởng trực tiếp đến độ nhạy và độ chính xác của hệ thống và cần được tinh chỉnh cẩn thận để đáp ứng yêu cầu hoạt động trong môi trường thực.

Mặc dù được đánh giá cao về độ chính xác trong nhiều trường hợp, một số nghiên cứu thực nghiệm đã chỉ ra rằng KDE có thể hoạt động không ổn định khi xử lý các tập dữ liệu có số chiều lớn hoặc cấu trúc phức tạp [38]. Điều này đặt ra nhu cầu phát triển các biến thể KDE hoặc tích hợp thêm các kỹ thuật hỗ trợ nhằm tăng tính khả thi và hiệu suất trong ứng dụng thực tế.

c. Phương pháp Centroid

Centroid (CEN) [38] được coi là một trong những phương pháp đơn giản nhất trong lĩnh vực phát hiện bất thường. Phương pháp này xây dựng một mô hình phát hiện bất thường mạng (NAD) từ dữ liệu huấn luyện bằng cách sử dụng hàm nhân Gauss (Gaussian kernel function).

Cụ thể, cho tập dữ liệu huấn luyện $X = \{x_1, x_2, ..., x_n\} \in R^d$, với n là số lượng điểm dữ liệu và d là số chiều (số thuộc tính). Đối với mỗi thuộc tính thứ j, chúng ta xác định giá trị trung bình μ_j và độ lệch chuẩn σ_j. Sau đó, tập dữ liệu X sẽ được chuẩn hóa (normalized) sử dụng chỉ số z (z-score). Chỉ số này được tính toán theo công thức 1.8. Quá trình chuẩn hóa này giúp đảm bảo rằng tất cả các thuộc tính đều có cùng thang đo, ngăn chặn các thuộc tính có giá trị lớn hơn chi phối quá trình tính toán khoảng cách hoặc mật độ trong mô hình.

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j} \tag{1.8}$$

trong đó x_{ij} là giá trị của thuộc tính thứ j của điểm dữ liệu x_i, và z_{ij} là chỉ số z tương ứng của nó. Thử nghiệm, khoảng cách Euclidean từ một điểm dữ liệu kiểm thử đến điểm dữ liệu trung tâm (centroid) của tập dữ liệu huấn luyện được tính toán và sử dụng làm độ đo bất thường (anomaly score) cho điểm dữ liệu đó. Các nghiên cứu đã chỉ ra rằng việc tích hợp (lai ghép) CEN với các phương pháp phát hiện bất thường khác thường mang lại hiệu quả tốt. Một ưu điểm đáng chú ý của CEN là nó là một trong số ít các phương pháp phát hiện bất thường không yêu cầu tham số, giúp đơn giản hóa quá trình triển khai và tối ưu

hóa. Mặc dù vậy, CEN hiếm khi được sử dụng như một phương pháp độc lập để phát hiện bất thường trong các nghiên cứu chuyên sâu, chủ yếu do hạn chế về khả năng nhận diện các kiểu bất thường phức tạp.

1.2.2. Phương pháp phân lớp đơn dựa trên học sâu

Học sâu (Deep Learning) là một nhánh quan trọng trong lĩnh vực học máy, đã và đang thu hút sự chú ý rộng rãi trong cộng đồng khoa học dữ liệu nhờ vào tiềm năng học biểu diễn mạnh mẽ từ dữ liệu có cấu trúc phức tạp. Không giống như các phương pháp học truyền thống, học sâu khai thác các kiến trúc mạng nơ-ron nhiều tầng (deep neural networks – DNNs), cho phép mô hình phân tích và trừu tượng hóa thông tin đầu vào qua nhiều cấp độ biểu diễn liên tiếp.

Về bản chất, mỗi tầng trong một mạng học sâu có thể học được một biểu diễn trung gian khác nhau, giúp mô hình hiểu sâu hơn về cấu trúc của dữ liệu. Điều này đặc biệt hiệu quả trong các bài toán có dữ liệu đầu vào có chiều cao hoặc có tính phi tuyến cao, như hình ảnh, chuỗi thời gian hoặc lưu lượng mạng. Một khảo sát gần đây [17] cho thấy rằng mặc dù học sâu không thể hiện ưu thế rõ ràng khi dữ liệu huấn luyện còn hạn chế, nhưng khi áp dụng trên các tập dữ liệu lớn, hiệu suất và khả năng khái quát của học sâu vượt trội so với các phương pháp truyền thống.

Một số kiến trúc học sâu tiêu biểu bao gồm Mạng nơ-ron tích chập (Convolutional Neural Networks – CNNs), có khả năng khai thác các đặc trưng không gian và mối quan hệ cục bộ trong dữ liệu thông qua các tầng lọc tích chập. CNN đặc biệt hiệu quả trong việc xử lý ảnh và dữ liệu chuỗi nhờ khả năng phát hiện các mẫu hình lặp lại ở nhiều mức độ trừu tượng.

Ngoài ra, Mạng đối kháng sinh (Generative Adversarial Networks – GANs) cũng là một hướng tiếp cận tiên tiến, trong đó hai mạng nơ-ron – gồm mô hình sinh (generator) và mô hình phân biệt (discriminator) – cùng tham gia huấn luyện theo cơ chế cạnh tranh. GAN đã mở rộng đáng kể khả năng của học

sâu, đặc biệt trong các ứng dụng tạo dữ liệu giả lập như hình ảnh, video hoặc văn bản, thường được gọi là “deepfake”.

Các mô hình học sâu có thể được chia thành ba loại chính:

1. Dạng học không có nhãn (generative learning models): Các mô hình trong nhóm này được huấn luyện từ dữ liệu đầu vào không gắn nhãn, với mục tiêu học ra phân bố tiềm ẩn hoặc cấu trúc nội tại của dữ liệu. Ví dụ điển hình là AutoEncoder hoặc các biến thể của GAN

2. Mô hình phân biệt (supervised hoặc discriminative learning modDạng học có nhãn (discriminative learning models): Đây là những mô hình học từ các cặp dữ liệu – nhãn nhằm mục đích tối ưu hóa khả năng phân loại hoặc dự đoán. CNN, RNN là đại diện tiêu biểu

3. Dạng kết hợp (hybrid models): Là sự pha trộn giữa khả năng học biểu diễn và phân biệt, cho phép mô hình vừa khai thác cấu trúc dữ liệu vừa tận dụng thông tin nhãn một cách hiệu quả

Trong ngữ cảnh phát hiện bất thường, một hướng nghiên cứu nổi bật là phân lớp đơn học sâu (Deep One-Class Classification – Deep OCC), tập trung vào việc phát hiện điểm dị thường trong tình huống chỉ có dữ liệu từ một lớp (thường là lớp bình thường). Các kiến trúc thường được sử dụng cho mục tiêu này gồm: Mạng niềm tin sâu (Deep Belief Network – DBN), Mạng nơ-ron hồi quy (Recurrent Neural Network – RNN), và đặc biệt là AutoEncoder (AE).

AutoEncoder là một kiến trúc học sâu truyền thẳng (feedforward neural network) hoạt động theo cơ chế học không giám sát, với mục tiêu chính là tái tạo dữ liệu đầu vào. Mạng bao gồm hai thành phần: bộ mã hóa (encoder) chịu trách nhiệm nén thông tin vào không gian ẩn, và bộ giải mã (decoder) tái tạo lại dữ liệu gốc từ biểu diễn đó. Điểm mạnh của AutoEncoder nằm ở khả năng phát hiện các điểm bất thường thông qua lỗi tái tạo – tức là độ sai khác giữa dữ liệu đầu vào và đầu ra sau giải mã.

Nhờ khả năng học đặc trưng tự động, khử nhiễu và định lượng sai lệch một cách trực tiếp, AutoEncoder đã chứng minh được tính hiệu quả cao trong phát hiện bất thường, đặc biệt trong các trường hợp dữ liệu có cấu trúc phức tạp hoặc số chiều lớn như lưu lượng mạng trong hệ thống an ninh thông tin..

a. Kiến trúc mạng nơ-ron AutoEncoder

AutoEncoder (AE) là một kiến trúc mạng nơ-ron nhân tạo thuộc nhóm học không giám sát, được thiết kế với mục tiêu học biểu diễn nén của dữ liệu đầu vào bằng cách tái tạo lại chính nó. Mạng AE bao gồm hai thành phần chính: bộ mã hóa (encoder) và bộ giải mã (decoder), hoạt động liên tiếp để thực hiện quá trình mã hóa và giải mã thông tin.

Cụ thể, với một đầu vào $x \in \mathbb{R}^n$, bộ mã hóa thực hiện biến đổi dữ liệu sang một không gian tiềm ẩn $z \in \mathbb{R}^m$ (với $m < n$), thông qua một hàm ánh xạ phi tuyến:

$$z = f\_\theta(x) = \sigma(Wx + b) \qquad (1.6)$$

Trong đó, W là ma trận trọng số, b là vector dịch, σ là hàm kích hoạt (activation function), và $\theta = \{W, b\}$ là tập tham số huấn luyện. Biểu diễn z chứa các đặc trưng ẩn của x đã được trừu tượng hóa.

Sau đó, bộ giải mã nhận đầu vào là z và cố gắng tái tạo lại dữ liệu ban đầu thông qua một hàm ánh xạ ngược:

$$\hat{x} = g\_\phi(z) = \sigma'(W'z + b') \qquad (1.7)$$

Trong đó, $\phi = \{W', b'\}$ là tập tham số của bộ giải mã. Mục tiêu huấn luyện AutoEncoder là tối thiểu hóa sai số giữa đầu vào gốc x và đầu ra tái tạo $\hat{x}$, thông qua một hàm mất mát điển hình như sai số bình phương:

$$L(x, \hat{x}) = \|x - \hat{x}\|^2 \qquad (1.8)$$

Trong quá trình huấn luyện, mạng học cách sao cho biểu diễn z giữ được thông tin quan trọng nhất để tái cấu trúc đầu vào. Khi áp dụng vào phát hiện bất thường, AutoEncoder được huấn luyện chỉ với dữ liệu bình thường. Trong giai đoạn kiểm thử, nếu một điểm dữ liệu mới tạo ra lỗi tái tạo lớn vượt ngưỡng cho phép, điểm đó được xem là bất thường.

Nhờ khả năng học biểu diễn tự động và nhận diện sai lệch thông qua lỗi tái tạo, AutoEncoder đã trở thành một công cụ hiệu quả và phổ biến trong các hệ thống phát hiện bất thường, đặc biệt trong các miền dữ liệu có số chiều lớn như lưu lượng mạng.

b. Phương pháp Shrink AutoEncoder (SAE)

Trong mô hình Stacked AutoEncoder (SAE), một bộ điều chuẩn (regularizer) được thêm vào hàm mất mát gốc của AutoEncoder nhằm định hướng quá trình học tập sao cho các biểu diễn ẩn (latent vectors) của dữ liệu bình thường hội tụ về gần gốc tọa độ trong không gian mã hoá. Quá trình huấn luyện SAE chỉ sử dụng dữ liệu thuộc lớp bình thường, và regularizer chịu trách nhiệm "kéo" các vector ẩn này về tâm, nhờ đó tăng cường khả năng dự đoán chính xác. Trong công trình của Cao cùng cộng sự [38], SAE được thử nghiệm trên nhiều bộ dữ liệu chuẩn trong lĩnh vực phát hiện bất thường mạng (NAD) và cho kết quả chính xác phát hiện cải thiện rõ rệt so với các phương pháp truyền thống. Hàm mất mát của SAE, xuất phát từ biểu thức (1.9) của AE, được mở rộng như sau:

$$Loss_{SAE}(\theta) = Loss_{RE}(\theta) + Regularizer(\theta) \qquad (1.10)$$

Trong biểu thức (1.10), thành phần đầu tiên biểu diễn độ lỗi tái tạo (Reconstruction Error – RE), phản ánh mức độ sai lệch giữa đầu vào và đầu ra được tái tạo bởi mô hình. Thành phần thứ hai là điều kiện điều chuẩn (regularization term), được thiết kế để dẫn hướng các vector biểu diễn ẩn tại tầng cổ chai hội tụ về gốc tọa độ trong không gian đặc trưng ẩn. Cụ thể, hàm

mục tiêu (loss function) của mô hình Stacked AutoEncoder (SAE) có thể được định nghĩa như sau:

$$Loss_{SAE}(\theta) = \frac{1}{m}\left(\sum_{i=1}^{m}(x_i - \hat{x_i})^2 + \alpha\sum_{i=1}^{m}\|z_i\|^2 \right. \tag{1.11}$$

trong đó m là lực lượng của tập huấn luyện; z_i và $\hat{x}_i$ tương ứng là vector lớp ẩn ứng với điểm dữ liệu quan sát x_i và giá trị tái tạo; tham số điều chỉnh mức độ cân bằng giữa hai thành phần của hàm mất mát α.

Mặc dù Stacked AutoEncoder (SAE) là một trong những mô hình học sâu nổi bật được ứng dụng rộng rãi trong phát hiện bất thường mạng máy tính, phương pháp này vẫn tồn tại một số hạn chế nhất định. Thứ nhất, SAE có xu hướng nén toàn bộ dữ liệu bình thường vào một cụm duy nhất trong không gian biểu diễn ẩn, do đó mô hình hoạt động không hiệu quả khi dữ liệu huấn luyện có cấu trúc phân cụm (multi-cluster). Thứ hai, mặc dù SAE đạt hiệu quả cao trong nhiều kịch bản, nhưng lại tỏ ra thiếu nhạy với một số loại tấn công, điển hình như kiểu tấn công Remote-to-Local (R2L). Các mẫu tấn công này khi được mã hóa bằng SAE thường có biểu diễn ẩn nằm gần gốc tọa độ – tương tự với dữ liệu bình thường – gây khó khăn trong việc phân biệt giữa hai lớp.

Nguyên nhân chính có thể xuất phát từ cơ chế hoạt động của SAE: mô hình được thiết kế để ép dữ liệu bình thường hội tụ về một vùng gần gốc tọa độ trong không gian đặc trưng ẩn. Do vậy, nếu dữ liệu tấn công có đặc điểm gần giống dữ liệu bình thường, SAE có xu hướng tái cấu trúc chúng về cùng vùng không gian, làm giảm hiệu quả phát hiện dị thường. Đây là một trong những hạn chế tiêu biểu của mô hình NAD dựa trên AutoEncoder trong việc nhận diện các dị thường tinh vi.

Tổng kết lại, phần này đã khảo sát các phương pháp phân lớp đơn tiêu biểu được ứng dụng cho bài toán NAD trong những năm gần đây. Kết quả cho thấy các phương pháp này, đặc biệt là các kỹ thuật học sâu, thể hiện tiềm năng

lớn trong bối cảnh dữ liệu ngày càng lớn và phức tạp. Tuy nhiên, ngay cả những mô hình tiên tiến như SAE vẫn chưa giải quyết triệt để được toàn bộ thách thức đặt ra. Do đó, nghiên cứu và cải tiến liên tục trong lĩnh vực NAD là yêu cầu tất yếu nhằm nâng cao năng lực phòng thủ trước các mối đe dọa mạng đang ngày một phát triển.

1.3. ĐỘ ĐO ĐÁNH GIÁ MÔ HÌNH PHÂN LỚP

Để đánh giá chất lượng thực nghiệm của một mô hình phát hiện bất thường, hai yếu tố then chốt cần được xem xét là tập dữ liệu kiểm thử và các tiêu chí đánh giá hiệu quả của mô hình.

Nhìn chung, các chỉ số đánh giá trong lĩnh vực phát hiện bất thường (AD) có thể được chia thành hai nhóm lớn:

- Chỉ số về hiệu năng (Performance Metrics): Phản ánh mức độ tiêu thụ tài nguyên của hệ thống trong quá trình vận hành thuật toán, bao gồm thời gian xử lý, mức sử dụng CPU và dung lượng bộ nhớ. Đây là các chỉ số có liên quan đến tính khả thi và chi phí triển khai trong môi trường thực tế.

- Chỉ số về hiệu quả (Effectiveness Metrics): Được sử dụng để đánh giá khả năng của mô hình trong việc phát hiện các điểm dữ liệu bất thường. Nhóm chỉ số này thường đo lường mức độ chính xác trong việc phân biệt giữa dữ liệu bình thường và bất thường, cũng như khả năng duy trì độ ổn định khi thử nghiệm với các tập dữ liệu khác nhau hoặc trong các điều kiện thay đổi.

Với sự tiến bộ nhanh chóng của công nghệ phần cứng — đặc biệt là sự phổ biến của các bộ xử lý hiệu năng cao — trọng tâm nghiên cứu hiện nay có xu hướng ưu tiên cải thiện độ chính xác và khả năng phát hiện hơn là tối ưu hóa chi phí tính toán.

Ngoài ra, tùy vào dạng đầu ra của hệ thống NAD (chẳng hạn: điểm số bất thường hoặc nhãn nhị phân), các chỉ số đánh giá sẽ được lựa chọn sao cho phù hợp nhằm đảm bảo phản ánh đúng đặc trưng của phương pháp được áp dụng.

1.3.1. Các chỉ số đánh giá dành cho đầu ra dạng nhãn nhị phân

Trong các mô hình học máy, độ chính xác (Accuracy – ACC) là một chỉ số quan trọng, được sử dụng rộng rãi để đánh giá khả năng phân loại đúng của hệ thống trên toàn bộ tập dữ liệu kiểm thử. Đối với các bài toán phân lớp đơn trong phát hiện bất thường, chỉ số này phản ánh mức độ mà mô hình có thể nhận diện chính xác cả các trường hợp "bình thường" lẫn "bất thường" trong tổng số mẫu đã kiểm tra.

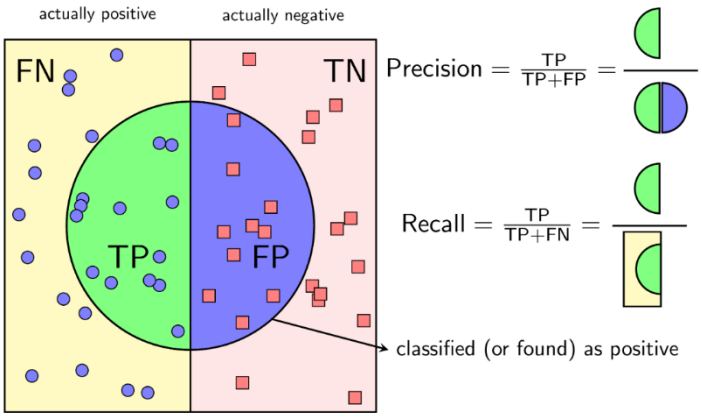
Cụ thể, độ chính xác được tính như tỷ lệ giữa số lượng mẫu được phân loại đúng và tổng số mẫu trong tập dữ liệu, cho thấy hiệu suất chung của mô hình khi xử lý đầu ra nhị phân. Công thức tính toán ACC được mô tả như sau:

$$Accuracy(ACC) = \frac{TP + TN}{TP + FP + FN + TN} \tag{1.12}$$

Trong quá trình đánh giá hiệu suất của mô hình phân lớp đơn, các khái niệm TP (True Positive), FP (False Positive), TN (True Negative) và FN (False Negative) được xác định từ ma trận nhầm lẫn (confusion matrix). Khi sử dụng độ chính xác (Accuracy – ACC) như một tiêu chí đánh giá, mô hình có giá trị ACC càng cao được coi là có năng lực phân loại tổng thể tốt hơn. Tuy nhiên, ACC chỉ phản ánh phần trăm mẫu được phân loại đúng, mà không thể hiện được sự phân bố của từng loại sai sót – chẳng hạn như phân loại sai mẫu bất thường là bình thường và ngược lại.

Để có cái nhìn đầy đủ hơn về chất lượng dự đoán, ma trận nhầm lẫn thường được sử dụng như một công cụ bổ trợ quan trọng. Trong ma trận này, các hàng thể hiện giá trị thực của dữ liệu (ground truth), trong khi các cột tương ứng với kết quả phân loại do mô hình dự đoán.

Đối với các mô hình phát hiện bất thường trong lĩnh vực an ninh mạng, nhiệm vụ chính là tách biệt giữa luồng dữ liệu mạng bình thường và luồng dữ liệu bất thường. Trong thiết lập phân lớp đơn, các mẫu thuộc lớp "bình thường" được gán là âm tính (negative), còn các mẫu bất thường – không thuộc vào lớp huấn luyện ban đầu – được xem là dương tính (positive). Cách tiếp cận này cho phép xác định các chỉ số đánh giá như Precision, Recall và F1-score, giúp phản ánh rõ ràng hơn hiệu quả của mô hình trên từng khía cạnh cụ thể của quá trình phát hiện.



Hình 1.4: Ma trận lỗi (Confusion Matrix).

Các chỉ số TP, FP, TN, FN là cơ sở để tính toán các tiêu chí đánh giá quan trọng như Accuracy, Precision, Recall, và F1-Score, giúp định lượng hiệu quả của hệ thống phát hiện bất thường trong thực tế.

Bên cạnh độ chính xác tổng thể (Accuracy – ACC), hai chỉ số quan trọng thường được sử dụng để đánh giá hiệu quả của mô hình phân lớp trong bài toán phát hiện bất thường là Tỷ lệ phát hiện (Detection Rate – DR) và Tỷ lệ cảnh báo sai (False Alarm Rate – FAR).

- Tỷ lệ phát hiện (Detection Rate – DR), còn được gọi là True Positive Rate (TPR), đo lường khả năng của hệ thống trong việc phát hiện chính xác các mẫu dữ liệu bất thường hoặc tấn công. DR được định nghĩa là tỷ lệ giữa số lượng tấn công được phát hiện đúng và tổng số tấn công thực tế trong tập dữ liệu.:

$$DR = \frac{TP}{TP + FN} \tag{1.13}$$

- Tỉ lệ báo động giả (False Alarm Rate - FAR) được định nghĩa là tỉ lệ giữa số lượng điểm dữ liệu bình thường bị hệ thống gắn cờ cảnh báo nhầm với tổng số điểm dữ liệu bình thường. Chỉ số FAR được tính theo công thức sau:

$$FAR = \frac{FP}{FP + TN} \tag{1.14}$$

Khi so sánh các mô hình phát hiện bất thường ở cùng một mức độ Tỷ lệ cảnh báo sai (False Alarm Rate – FAR), mô hình nào đạt Tỷ lệ phát hiện (Detection Rate – DR) cao hơn sẽ được xem là có hiệu suất phát hiện tốt hơn. Tuy nhiên, việc đánh giá mô hình dựa trên chỉ số DR hoặc FAR đơn lẻ có thể chưa phản ánh đầy đủ hiệu quả tổng thể, đặc biệt trong các bài toán có phân bố dữ liệu không cân bằng.

Trong các kịch bản thực tế, chẳng hạn như phát hiện bất thường mạng, dữ liệu thường bị mất cân đối nghiêm trọng giữa số lượng mẫu bình thường và bất thường. Quan trọng hơn, việc phát hiện sai các mẫu bất thường (false negatives) thường có tác động nghiêm trọng hơn so với việc dự đoán sai các mẫu bình thường. Do đó, các chỉ số như Accuracy (ACC), Detection Rate (DR) và False Alarm Rate (FAR) có thể không đủ nhạy trong việc đánh giá chính xác hiệu năng mô hình trong bối cảnh này [7].

Để khắc phục các hạn chế nói trên, chỉ số F1-Score được đề xuất như một độ đo thay thế hiệu quả hơn. F1-Score là trung bình điều hòa (harmonic mean) giữa Precision (tỉ lệ đúng trong các cảnh báo dương tính) và Recall (chính là DR – khả năng phát hiện bất thường). F1-score đặc biệt phù hợp để đánh giá các mô hình trong môi trường dữ liệu không cân bằng, giúp cân bằng giữa khả năng phát hiện và mức độ chính xác của cảnh báo.

- **Precision** (Độ chính xác dương tính) được định nghĩa là tỷ lệ giữa số lượng mẫu dương tính thật sự (True Positive – TP) và tổng số mẫu được mô

hình phân loại là dương tính (bao gồm cả TP và False Positive – FP). Precision phản ánh mức độ chính xác của mô hình khi đưa ra dự đoán bất thường.

$$precision = \frac{TP}{TP + FP} \tag{1.15}$$

Recall (Tỷ lệ phát hiện hay Độ nhạy) được định nghĩa là tỷ lệ giữa số lượng mẫu dương tính thật sự (TP) và tổng số mẫu thực sự thuộc lớp dương tính (gồm TP và False Negative – FN). Recall phản ánh khả năng của mô hình trong việc phát hiện các bất thường hiện diện trong dữ liệu.

$$recall = \frac{TP}{TP + FN} \tag{1.16}$$

Công thức tính F1-score như sau:

$$F1 - score = 2\frac{1}{\frac{1}{precision} + \frac{1}{recall}} = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{1.17}$$

F1-score là một độ đo tổng hợp, phản ánh sự cân bằng giữa hai chỉ số Precision và Recall trong đánh giá hiệu quả của mô hình học máy. Với bài toán phát hiện bất thường có 2 lớp, F1-score được xem là chỉ số đánh giá quan trọng hàng đầu, đặc biệt trong các tình huống dữ liệu không cân bằng.

Giá trị F1-score càng cao cho thấy mô hình không chỉ có khả năng phát hiện chính xác (recall cao) mà còn đưa ra ít cảnh báo sai (precision cao), từ đó phản ánh năng lực nhận diện bất thường một cách hiệu quả và đáng tin cậy hơn.

1.3.2. Chỉ số đánh giá với đầu ra là điểm số bất thường

Đối với các phương pháp phát hiện bất thường mà đầu ra là một giá trị liên tục biểu thị mức độ bất thường (anomaly score), thay vì nhãn nhị phân, việc đánh giá hiệu quả mô hình đòi hỏi các công cụ phân tích chuyên biệt hơn. Trong trường hợp này, đường cong ROC (Receiver Operating Characteristic) và diện tích dưới đường cong ROC (Area Under the Curve – AUC) là hai chỉ số đánh giá được sử dụng rộng rãi.

Cụ thể, đường cong ROC là một biểu đồ thể hiện khả năng phân biệt giữa hai loại dữ liệu – bình thường và bất thường – ở các ngưỡng phân loại khác nhau. Đường cong này vẽ mối quan hệ giữa tỷ lệ phát hiện (True Positive Rate – TPR) và tỷ lệ cảnh báo sai (False Positive Rate – FPR). Việc thay đổi ngưỡng áp dụng cho điểm số bất thường sẽ làm thay đổi các giá trị TPR và FPR, qua đó tạo thành một đường cong ROC mô tả toàn bộ phổ khả năng phân loại của mô hình.

Chỉ số AUC đóng vai trò là đại lượng định lượng cho chất lượng của đường cong ROC. Giá trị AUC càng tiến gần đến 1 cho thấy mô hình có khả năng phân biệt tốt giữa hai lớp. Ngược lại, AUC xấp xỉ 0.5 cho thấy mô hình không tốt hơn việc phân loại ngẫu nhiên.

Việc sử dụng ROC và AUC đặc biệt phù hợp trong các tình huống mà ranh giới phân loại không rõ ràng và cần đánh giá hiệu quả mô hình trên toàn bộ dải ngưỡng, thay vì một giá trị cố định.

Mỗi điểm trên đường cong ROC tương ứng với một ngưỡng cụ thể, từ đó xác định được cặp giá trị (DR, FAR) của mô hình tại ngưỡng đó. ROC cho phép đánh giá mức độ cân bằng giữa khả năng phát hiện tấn công và mức cảnh báo sai mà hệ thống tạo ra. Giá trị AUC – tức diện tích dưới đường cong ROC – cung cấp một đại lượng tổng quát hóa mức độ hiệu quả của mô hình; AUC gần 1 cho thấy hiệu suất phân loại cao, trong khi AUC xấp xỉ 0.5 phản ánh hiệu quả ngẫu nhiên.

$$ROC = \frac{P(x|positive)}{P(x|negaitive)} \tag{1.18}$$

Trong biểu diễn đường cong ROC (Receiver Operating Characteristic), đỉnh lý tưởng của đường cong nằm gần góc tọa độ [0, 1], phản ánh mô hình có thể phát hiện các bất thường trong dữ liệu với Tỷ lệ phát hiện (True Positive Rate – TPR) cao và Tỷ lệ cảnh báo sai (False Positive Rate – FPR) thấp. Mô

hình có đường cong ROC càng tiến gần đến góc này được xem là hoạt động càng hiệu quả.

Chỉ số AUC (Area Under the Curve) biểu thị diện tích dưới đường cong ROC và được xem là thước đo tổng quát cho khả năng phân biệt giữa hai lớp dữ liệu của mô hình học máy. AUC phản ánh hiệu suất phân loại trung bình trên toàn bộ các ngưỡng phân tách có thể có. Một mô hình có giá trị AUC tiến gần về 1 cho thấy khả năng phân biệt các điểm dữ liệu bình thường và bất thường một cách mạnh mẽ và ổn định.

Chỉ số AUC đặc biệt hữu ích trong các trường hợp mô hình đưa ra đầu ra dạng điểm số (anomaly score) thay vì nhãn nhị phân cố định, do đó chưa thể xác định ngưỡng quyết định cụ thể. Nhờ tính độc lập với ngưỡng và khả năng so sánh giữa các mô hình, AUC được sử dụng trong việc đánh giá và lựa chọn thuật toán cho các hệ thống phát hiện bất thường.

1.3.3. Độ ổn định của mô hình

Độ ổn định (stability) của mô hình trên các môi trường mạng khác nhau được xem là một tiêu chí đánh giá quan trọng trong các hệ thống phát hiện bất thường mạng. Một mô hình có thể được xem là đáng tin cậy nếu duy trì được hiệu suất phát hiện cao trên nhiều tập dữ liệu có đặc điểm khác nhau.

Cụ thể, khi so sánh hiệu quả giữa các mô hình NAD, nếu một giải thuật thể hiện các chỉ số đánh giá như F1-score hoặc Accuracy (ACC) có độ biến thiên thấp và duy trì hiệu năng ổn định trên nhiều bộ dữ liệu kiểm thử khác nhau, khi đó mô hình được đánh giá là có tính tổng quát và khả năng thích ứng cao hơn so với các mô hình còn lại. Tính ổn định này đặc biệt quan trọng trong bối cảnh các môi trường mạng ngày càng đa dạng và có hành vi dữ liệu phức tạp.

1.4. KẾT LUẬN

Chương này tập trung trình bày khái niệm bất thường trong mạng máy tính, đồng thời tổng quan các kỹ thuật phát hiện bất thường ở mức độ tổng quát và các phương pháp tiếp cận dựa trên phân tích lưu lượng mạng. Nội dung được đề cập bao gồm phân loại các kỹ thuật truyền thống như dựa trên luật, thống kê, cũng như các phương pháp hiện đại sử dụng học máy và học sâu. Bên cạnh đó, chương còn giới thiệu các chỉ số đánh giá hiệu suất thường dùng nhằm đo lường mức độ chính xác, độ bao phủ và tỷ lệ phát hiện của hệ thống phát hiện bất thường. Những nội dung này mang lại cái nhìn tổng quát và cập nhật về lĩnh vực đang nghiên cứu, giúp hiểu rõ các hướng tiếp cận phổ biến cũng như đặc điểm của từng phương pháp trong việc xử lý bài toán phát hiện bất thường trong dữ liệu mạng hiện nay.

CHƯƠNG 2. LÝ THUYẾT TẬP MỜ, ĐẠI SỐ GIA TỬ VÀ PHƯƠNG PHÁP XÂY DỰNG HỆ PHÂN LỚP MỜ

Nội dung chương này trình bày các cơ sở lý thuyết nền tảng cho việc xây dựng mô hình phát hiện bất thường mạng máy tính bằng hệ thống suy diễn mờ (FRBS) kết hợp với đại số gia tử (HA). Mở đầu chương là phần tổng quan về lý thuyết tập mờ, mô tả cách thức biểu diễn các giá trị không chắc chắn và các hàm thành viên tương ứng. Sau đó, chương đi sâu vào lý thuyết đại số gia tử (HA), một công cụ toán học dùng để định lượng các giá trị ngôn ngữ, giúp mô hình hóa các khái niệm mờ một cách có hệ thống và có thứ tự ngữ nghĩa.

Hệ thống suy diễn mờ dựa trên luật (FRBS) vốn có ưu điểm trong xử lý dữ liệu mơ hồ và ra quyết định tương tự con người, tuy nhiên khi áp dụng trong bài toán phân lớp với dữ liệu phức tạp, khả năng định lượng các khái niệm ngôn ngữ và xây dựng luật mờ vẫn còn hạn chế. Đại số gia tử (Hedge Algebra – HA) được tích hợp như một công cụ bổ sung mạnh mẽ, giúp mô hình hóa ngữ nghĩa của các từ ngôn ngữ một cách định lượng, chính xác và có hệ thống hơn.

Việc kết hợp HA vào quá trình xây dựng FRBS cho phép tự động sinh luật mờ dựa trên cấu trúc đại số của ngôn ngữ, khắc phục điểm yếu về tính chủ quan trong việc thiết lập luật thủ công ở hệ mờ truyền thống. Ngoài ra, HA hỗ trợ khả năng phân hoạch mờ tối ưu (OPHA), từ đó nâng cao hiệu quả trong việc phân biệt giữa dữ liệu bình thường và bất thường trong mạng. Sự kết hợp này vì vậy không chỉ nâng cao độ chính xác của hệ thống phát hiện bất thường mà còn đảm bảo tính khả thi khi triển khai thực tế trong môi trường mạng động và phức tạp.

2.1. CƠ BẢN VỀ LÝ THUYẾT TẬP MỜ

Lý thuyết tập mờ (Fuzzy Set Theory), do Lotfi A. Zadeh đề xuất vào năm 1965 [39], đã mở rộng khái niệm truyền thống của tập hợp trong toán học.

Trong lý thuyết này, thay vì một phần tử chỉ có thể hoàn toàn thuộc hoặc không thuộc một tập hợp như trong lý thuyết tập hợp cổ điển, tập mờ cho phép biểu diễn mức độ thuộc theo cách liên tục.

Cụ thể, giả sử U là tập vũ trụ gồm các phần tử, một tập mờ A được định nghĩa thông qua một hàm thành viên $\mu_A: U \rightarrow [0, 1]$, trong đó $\mu_A(x)$ thể hiện mức độ mà phần tử x thuộc vào tập A. Khi giá trị $\mu_A(x)$ càng gần 1, mức độ thuộc của x vào A càng cao; ngược lại, giá trị gần 0 biểu thị mức độ không thuộc.

Điểm khác biệt then chốt giữa tập cổ điển và tập mờ nằm ở giá trị của hàm đặc trưng: trong khi tập cổ điển chỉ cho phép giá trị 0 hoặc 1 (thuộc hoặc không thuộc), thì tập mờ cung cấp một phổ liên tục, cho phép biểu diễn các mối quan hệ không rõ ràng, mơ hồ hoặc trung gian.

Cách tiếp cận này đã mở ra khả năng mô hình hóa và xử lý các tình huống chứa thông tin không chắc chắn hoặc chưa đầy đủ, từ đó trở thành nền tảng quan trọng trong nhiều ứng dụng như điều khiển mờ, hệ chuyên gia, và xử lý ngôn ngữ tự nhiên..

2.1.1. Định nghĩa tập mờ

Định nghĩa 2.1: [39]

Giả sử U là một tập hợp chứa các phần tử x, tức $U = \{x\}$. Một tập mờ B trên U được định nghĩa là một tập hợp các cặp $(x, \mu_B(x))$, trong đó $x \in U$ và μ_B là một ánh xạ từ U vào đoạn [0, 1]:

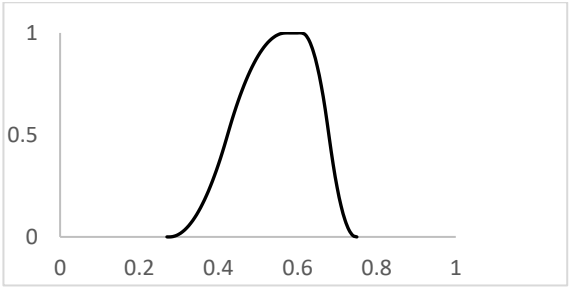
$$\mu_B: U \rightarrow [0, 1] \qquad (2.1)$$

Ánh xạ μ_B được gọi là hàm thành viên (membership function) của tập mờ B. Tập hợp U được coi là tập nền (universe of discourse) của B. Tập mờ B được ký hiệu là:

$$B = \{(x, \mu_B(x)) \mid x \in U\}$$

Giá trị của μ_B(x) thể hiện mức độ mà phần tử x thuộc vào tập mờ B. Khi μ_B(x) tiến gần đến 1, điều đó cho thấy x có mức độ thuộc về B cao hơn. Ngược lại, nếu μ_B(x) gần 0, mức độ thuộc của x vào B là thấp.

Tập mờ là một sự mở rộng khái niệm của tập hợp cổ điển. Trong trường hợp tập hợp cổ điển, hàm đặc trưng μ_B(x) chỉ nhận các giá trị rời rạc: 1 nếu x thuộc B và 0 nếu không thuộc. Tuy nhiên, trong tập mờ, hàm thành viên có thể nhận bất kỳ giá trị thực nào trong khoảng [0, 1], qua đó cho phép mô hình hóa các trạng thái "thuộc một phần" – điều không thể hiện diện trong lý thuyết tập hợp cổ điển.. Hình 2.1 ví dụ đồ thị của một hàm thành viên.



Hình 2.1. Một hàm thành viên dạng hình thang của tập mờ A

2.1.2. Xây dựng hàm thành viên

Khi xây dựng hàm thành viên (membership function) cho một tập mờ B, giá trị của hàm phải nằm trong khoảng [0, 1] để phản ánh mức độ thuộc của từng phần tử trong tập vũ trụ. Trên thực tế, tùy thuộc vào đặc thù của bài toán và tính chất dữ liệu, có một số dạng hàm thành viên phổ biến thường được áp dụng trong lý thuyết tập mờ, bao gồm: hàm tam giác (triangular), hàm hình thang (trapezoidal), và hàm Gaussian. Mỗi dạng hàm mang đến mức độ linh hoạt riêng trong việc biểu diễn tính mờ.

- Hàm thành viên dạng tam giác: $\mu_B(x) = \max(\min\left(\frac{x-a1}{b2-a1}, \frac{a3-x}{a3-a2}\right), 0)$, trong đó a_1, a_2, a_3 lần lượt là chân bên trái, đỉnh và chân bên phải của tam giác.

- Hàm thành viên dạng hình thang: $\mu_B(x) = \max(\min\left(\frac{x-a1}{a2-a1}, \frac{a4-x}{a4-a3}, 1\right), 0)$, trong đó a_1, a_4 lần lượt là là đỉnh dưới bên trái, bên phải, a_2, a_3 lần lượt là đỉnh trên bên trái, bên phải của hình thang.

- Hàm thành viên Gauss: $\mu_B(x) = e^{-\frac{(a-x)^2}{2c^2}}$, trong đó c là độ rộng và a vị trí đỉnh của hàm.

Trong số các dạng hàm thành viên được sử dụng để biểu diễn tập mờ, hàm tam giác (triangular membership function) là loại được áp dụng phổ biến nhất trong thực tiễn. Sở dĩ như vậy là vì hàm này có cấu trúc hình học đơn giản, dễ triển khai và trực quan đối với người dùng. Việc xác định tham số của hàm tam giác cũng không phức tạp, phù hợp với các hệ thống suy luận mờ đòi hỏi hiệu năng tính toán cao mà vẫn đảm bảo khả năng diễn giải tốt..

2.1.3. Biến ngôn ngữ

Theo khái niệm được đề xuất bởi Zadeh [40], một biến ngôn ngữ (linguistic variable) là loại biến đặc biệt mà giá trị của nó không phải là những con số chính xác, mà là các thuật ngữ ngôn ngữ – những từ hoặc cụm từ có ý nghĩa định tính, được biểu diễn trong ngôn ngữ tự nhiên hoặc ngôn ngữ hình thức.

Chẳng hạn, xét thuộc tính chiều cao của trẻ nhỏ, có thể định nghĩa một biến ngôn ngữ mang tên “Chiều cao” (Height). Biến này có thể nhận các giá trị ngôn ngữ như “Rất thấp”, “Thấp”, “Trung bình”, “Cao”, v.v. Mỗi giá trị ngôn ngữ được gắn với một hàm thành viên (membership function), nhằm định lượng mức độ mà một giá trị cụ thể (ví dụ: chiều cao đo được) phù hợp với mô tả ngôn ngữ tương ứng.

Giả sử miền giá trị của chiều cao nằm trong khoảng từ 0.5m đến 1.2m. Các giá trị ngôn ngữ kể trên sẽ được sinh ra từ một tập các hàm thành viên, có thể được thiết kế thủ công hoặc học từ dữ liệu huấn luyện.

Định nghĩa 2.1: [40]

Một biến ngôn ngữ L được đặc trưng bởi bộ bốn thành phần L = (T(L), U, G, M), trong đó:

- T(L): Là tập hợp các thuật ngữ ngôn ngữ (term set) mà biến L có thể nhận, ví dụ: Rất thấp, Thấp, Trung bình, Cao.

- U: Là tập nền (universe of discourse), có thể là rời rạc hoặc liên tục, biểu diễn không gian giá trị số thực tế của thuộc tính, ví dụ: [0.5, 1.2] cho chiều cao.

- G: Là tập các quy tắc cú pháp (syntactic rules), dùng để tạo ra những giá trị hợp lệ trong T(L).

- M: Là tập các ánh xạ (mappings) liên kết mỗi thuật ngữ trong T(L) với một hàm thành viên cụ thể xác định trên không gian U.

Trên nền tảng lý thuyết tập mờ và khái niệm biến ngôn ngữ, lĩnh vực lập luận xấp xỉ (approximate reasoning) đã được phát triển nhằm mô phỏng cách con người suy luận với thông tin không rõ ràng hoặc không đầy đủ. Cốt lõi của lý thuyết này là khả năng đưa ra kết luận dựa trên biểu diễn ngôn ngữ kết hợp với mức độ không chắc chắn.

Một trong những hướng ứng dụng nổi bật là mô hình hệ thống mờ dựa trên luật. Hệ này sử dụng tập các luật IF–THEN dạng mờ để mô hình hóa tri thức chuyên gia, đi kèm với cơ chế suy diễn mờ để đưa ra kết luận từ dữ liệu đầu vào. Đặc điểm nổi bật của mô hình là khả năng diễn giải rõ ràng, dễ điều chỉnh và đặc biệt phù hợp với các tình huống ra quyết định trong môi trường không chắc chắn.

2.1.4. Phân hoạch mờ

Phân hoạch mờ (fuzzy partitioning) là một khái niệm cốt lõi trong lý thuyết tập mờ, được sử dụng để chuyển miền giá trị của một biến ngôn ngữ thành các vùng mờ chồng lấp nhau. Mỗi vùng được xác định thông qua một hàm thành

viên cụ thể, cho phép một giá trị trong miền đầu vào có thể thuộc về nhiều tập mờ với các mức độ khác nhau.

Mục tiêu chính của phân hoạch mờ là hỗ trợ việc biểu diễn dữ liệu số dưới dạng ngôn ngữ mờ, phục vụ cho quá trình suy luận trong các hệ thống mờ. Một phân hoạch mờ hiệu quả phải đảm bảo tính liên tục (continuity), bao phủ đầy đủ không gian giá trị (completeness), đồng thời cho phép mô hình hóa tốt các tính chất không chắc chắn vốn có trong dữ liệu thực tế.

Giả sử U = [c, d] là không gian nền liên tục trong tập số thực R, với n điểm cố định $x_1 < x_2 < ... < x_n$ thuộc U, đóng vai trò là các điểm mốc. Một tập gồm m tập mờ $Y = \{B_1, B_2, ..., B_m\}$, với các hàm thành viên tương ứng μ_{B_1}, μ_{B_2}, ..., μ_{Bm}, tạo thành một phân hoạch mờ của U nếu thỏa mãn các điều kiện sau với mọi k = 1, ..., m:

1) $\mu_{Bk}(x_k) = 1$

2) Nếu $x \notin [x_{k-1}, x_{k+1}]$ thì $\mu_{Bk}(x) = 0$

3) $\mu_{Bk}(x)$ là hàm liên tục trên U

4) $\mu_{Bk}(x)$ đơn điệu tăng trên đoạn $[x_{k-1}, x_k]$ và đơn điệu giảm trên đoạn $[x_k, x_{k+1}]$

5) Với mọi $x \in U$, tồn tại ít nhất một k sao cho $\mu_{Bk}(x) > 0$

Nếu phân hoạch mờ còn thỏa mãn điều kiện:

6) $\sum_{k=1}^{m} \mu_{Bk}(x) = 1$ với mọi $x \in U$

thì được gọi là phân hoạch mờ mạnh (strong fuzzy partition).

Trong trường hợp phân hoạch còn thỏa mãn thêm:

7) Khoảng cách giữa các điểm mốc là hằng số: $h_k = x_{k+1} - x_k$ là không đổi với mọi $k \neq m$

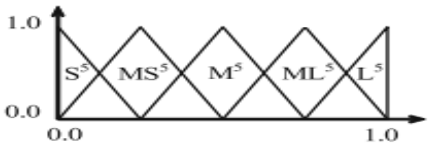
8) Các hàm thành viên là đối xứng

9) Các hàm thành viên có hình dạng hình học giống nhau

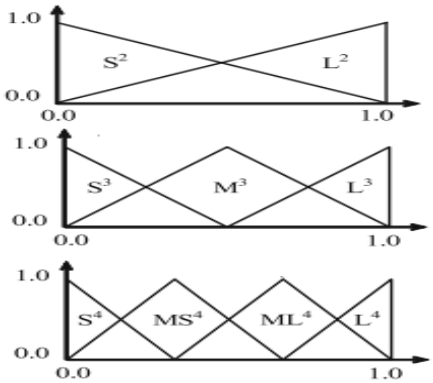
thì phân hoạch đó được gọi là phân hoạch mờ đều (uniform fuzzy partition).

Theo định nghĩa 2.3, mỗi phân hoạch mờ có thể được xem như một mức thể hạt (granularity) trong không gian thuộc tính. Nếu phân hoạch chỉ tồn tại ở một cấp độ chi tiết, ta gọi đó là phân hoạch mờ đơn thể hạt (single granularity). Ngược lại, nếu bao gồm nhiều cấp độ chi tiết tương ứng với nhiều thể hạt khác nhau, thì được gọi là phân hoạch mờ đa thể hạt (multi granularity).

Việc lựa chọn cấp độ thể hạt phù hợp có ảnh hưởng lớn đến hiệu quả biểu diễn tri thức và suy luận mờ, bởi mỗi mức granularity cho phép hệ thống tập trung vào những khía cạnh ngữ nghĩa khác nhau của dữ liệu, từ đó cải thiện tính chính xác và khả năng giải thích của mô hình.



Hình 2.2. Cấu trúc phân hoạch đơn thể hạt [14]



Hình 2.3. Cấu trúc phân hoạch đa thể hạt [14]

2.2. CƠ BẢN VỀ ĐẠI SỐ GIA TỬ

Kể từ khi được giới thiệu, lý thuyết tập mờ đã trải qua quá trình phát triển mạnh mẽ và được ứng dụng rộng rãi trong nhiều lĩnh vực, với mục tiêu chính là xây dựng các mô hình có khả năng mô phỏng cách con người suy luận trong

điều kiện không chắc chắn. Tuy vậy, lý thuyết này vẫn còn tồn tại một số hạn chế, đặc biệt trong việc diễn đạt một cách đầy đủ các khía cạnh ngữ nghĩa và cấu trúc của miền giá trị ngôn ngữ – yếu tố then chốt trong giao tiếp và lập luận tự nhiên. Trên thực tế, cho đến nay vẫn chưa tồn tại một khung lý thuyết hình thức nào có thể mô tả một cách nhất quán và định lượng được ngữ nghĩa của các thuật ngữ ngôn ngữ một cách toàn diện.

Nhằm khắc phục hạn chế này, vào năm 1990, Nguyễn Cát Hồ và W. Wechler [29, 30] đã đề xuất một cách tiếp cận mới, dựa trên cấu trúc ngữ nghĩa tự nhiên của miền giá trị của các biến ngôn ngữ. Các tác giả nhận thấy rằng tập giá trị của biến ngôn ngữ thường tồn tại quan hệ thứ tự ngữ nghĩa nội tại — ví dụ: "chậm" nhỏ hơn "nhanh", hay "ngắn" nhỏ hơn "dài" theo nghĩa ngôn ngữ học. Dựa trên mối quan hệ này, họ xây dựng nên một mô hình đại số có cấu trúc — gọi là đại số gia tử (Hedge Algebra – HA) — trong đó miền giá trị của biến ngôn ngữ được xem như một tập hợp có cấu trúc đại số.

Lý thuyết HA đưa ra một phương pháp tiếp cận theo hướng đại số nhằm mô hình hóa cấu trúc ngữ nghĩa bên trong của các thuật ngữ trong biến ngôn ngữ. Nó thiết lập một khung lý thuyết chặt chẽ cho việc kết nối giữa ngữ nghĩa định lượng – thường thấy trong lý thuyết tập mờ – và ngữ nghĩa ngữ pháp vốn có của ngôn ngữ tự nhiên. Mô hình này được xây dựng dựa trên nền tảng logic tiên đề và hình thức toán học, giúp nâng cao khả năng biểu diễn và xử lý thông tin mơ hồ hoặc không chắc chắn trong các hệ thống suy luận mờ một cách chính xác và có hệ thống hơn.

2.2.1. Khái niệm đại số gia tử

Đại số gia tử (tiếng Anh: Hedge Algebra – viết tắt: HA) là một lý thuyết toán học dùng để mô hình hóa và định lượng các giá trị ngôn ngữ mờ như "rất cao", "khá thấp", "trung bình", v.v... một cách có hệ thống và chính xác. Không giống như các phương pháp mờ truyền thống vốn chỉ gán giá trị cho từ ngữ một

cách cảm tính thông qua hàm thành viên, HA cho phép xây dựng một cấu trúc đại số dựa trên các toán tử tăng cường (hedges) và thứ tự ngữ nghĩa giữa các từ. Nhờ vậy, HA mang lại khả năng tự động hóa việc sinh luật, phân hoạch và định nghĩa các mức độ mờ, đồng thời phản ánh được mức độ tin cậy hoặc cường độ của các giá trị ngôn ngữ theo cách tương tự tư duy của con người.

Việc áp dụng HA đặc biệt hữu ích trong các hệ thống cần suy luận dựa trên dữ liệu ngôn ngữ không chắc chắn, như chẩn đoán, phân loại hoặc cảnh báo – điển hình là trong bài toán phát hiện bất thường mạng, nơi các trạng thái của lưu lượng mạng thường không rõ ràng và khó xác định ranh giới dứt khoát.

Định nghĩa 2.2 [32]: Một hàm HA (Hedge Algebra) được xác định như một bộ 4 thành phần và được ký hiệu là AX = (X, G, H, ≤), trong đó:

- X: Là tập hợp các thuật ngữ ngôn ngữ (terms).

- G: Là tập hợp các phần tử sinh (generators), đóng vai trò như các thành phần cơ sở để xây dựng nên các thuật ngữ ngôn ngữ khác. Tập G thường bao gồm các phần tử đặc biệt sau:

+ 0: Đại diện cho giá trị nhỏ nhất, thường mang ý nghĩa cực tiểu hoặc vắng mặt.

+ 1: Đại diện cho giá trị lớn nhất, thường thể hiện mức độ tối đa hoặc đầy đủ.

+ W: Là phần tử trung hòa (neutral), biểu thị giá trị trung bình hoặc trạng thái không ưu tiên.

- H: Là tập các gia tử (hedges), tức là các phép toán ngôn ngữ dùng để điều chỉnh hoặc biến đổi mức độ ngữ nghĩa của một thuật ngữ. Các gia tử cho phép tạo ra nhiều sắc thái nghĩa khác nhau như "rất", "hơi", "khá", v.v.

- ≤: Là quan hệ thứ tự ngữ nghĩa (semantic order) được xác lập trên tập X, phản ánh cách các thuật ngữ liên kết với nhau về mặt ngữ nghĩa.

Tập H bao gồm hai tập con:

- H^-: Là tập các gia tử âm (negative hedges), làm giảm cường độ ngữ nghĩa. $H^- = \{h_{-1} < h_{-2} < ... < h_{-q}\}$, ví dụ: “slightly”, “less”, “somewhat”.

- H^+: Là tập các gia tử dương (positive hedges), làm tăng cường độ ngữ nghĩa. $H^+ = \{h_1 < h_2 < ... < h_p\}$, ví dụ: “very”, “more”, “extremely”.

Khi áp dụng một gia tử $h \in H$ lên một phần tử $u \in X$, ta thu được thuật ngữ mới ký hiệu là hu. Với mỗi $u \in X$, H(u) là tập hợp các từ được tạo thành từ u bằng cách áp dụng các gia tử trong H. Nếu $u = h_n \ldots h_1 x$ với $h_n, \ldots, h_1 \in H$, $n \geq 1$, thì biểu thức này gọi là biểu diễn chính tắc của u.

Biểu thức hn…h1x là biểu diễn chính tắc nếu mỗi gia tử thực sự thay đổi ngữ nghĩa, tức là với mọi $i \leq n$, $hi \ldots h1x \neq hi_{-1} \ldots h1x$. Độ dài của một thuật ngữ ngôn ngữ u, ký hiệu là l(u), được định nghĩa là số lượng gia tử xuất hiện trong biểu diễn chính tắc của nó, cộng thêm một.

2.2.2. Tính chất của HA tuyến tính

Định lý 2.1: [32] Cho tập $\boldsymbol{H}^-$ và $\boldsymbol{H}^+$ là các tập sắp thứ tự tuyến tính của HA AX = (**X**, **G**, **H**, ≤). Khi đó ta có:

i) Với mỗi $x \in \boldsymbol{X}$ thì $\boldsymbol{H}(x)$ là tập sắp thứ tự tuyến tính.

ii) Nếu $\boldsymbol{X}$ được sinh từ $\boldsymbol{G}$ bởi các gia tử và $\boldsymbol{G}$ là tập sắp thứ tự tuyến tính thì $\boldsymbol{X}$ cũng là tập sắp thứ tự tuyến tính. Hơn nữa nếu $x < y$, và x, y là độc lập với nhau, tức là $x \notin \boldsymbol{H}(y)$ và $y \notin \boldsymbol{H}(x)$, thì $\boldsymbol{H}(x) \leq \boldsymbol{H}(y)$.

Định lý dưới đây trình bày cơ sở lý thuyết cho việc so sánh thứ tự ngữ nghĩa giữa hai từ ngôn ngữ bất kỳ trong miền giá trị ngôn ngữ của biến **X**. Cụ thể, định lý khai thác cấu trúc thứ tự tuyến tính được xác lập trong đại số gia tử, cho phép xác định mối quan hệ ngữ nghĩa giữa các từ ngôn ngữ sinh ra từ các phần tử sinh khác nhau (hoặc cùng một phần tử) thông qua các tổ hợp gia tử.

Định lý 2.2: [32] Định lý 2.2 [32]: Giả sử $u = h_n \ldots h_1 x$ và $v = k_m \ldots k_1 x$ là hai biểu diễn chính tắc tương ứng của các thuật ngữ ngôn ngữ u và v dựa trên phần tử sinh x. Khi đó, tồn tại một chỉ số j sao cho $j \leq \min\{n, m\} + 1$ và các gia tử tại những vị trí trước j là giống nhau, tức là $h_{j'} = k_{j'}$ với mọi $j' < j$.

Trong trường hợp $j = \min\{n, m\} + 1$, điều kiện được mở rộng như sau: nếu $n + 1 \leq m$ thì h_j là toán tử đơn vị I, hoặc nếu $m + 1 \leq n$ thì $k_j = I$.

Khi đó, quan hệ ngữ nghĩa giữa u và v được xác định theo các điều kiện sau:

i) $u < v$ khi và chỉ khi $h_j(u_j) < k_j(v_j)$, trong đó $u_j = h_{j-1} \ldots h_1 x$ là phần còn lại của biểu thức u tính đến vị trí j - 1.

ii) $u = v$ nếu và chỉ nếu $n = m$ và với mọi chỉ số, $h_j(u_j) = k_j(v_j)$.

iii) u và v không thể so sánh với nhau nếu và chỉ nếu các biểu thức $h_j(u_j)$ và $k_j(v_j)$ không thể thiết lập được quan hệ thứ tự giữa chúng.

Định lý này giúp xác định rõ ràng quan hệ thứ tự giữa các thuật ngữ ngôn ngữ thông qua cấu trúc của biểu diễn chính tắc và các gia tử tạo thành chúng..

2.2.3. Độ đo tính mờ của các từ ngôn ngữ

Khái niệm đo lường mức độ mờ (fuzziness) của một từ ngôn ngữ là một vấn đề phức tạp và không dễ xác định bằng trực giác. Do đó, đã có nhiều phương pháp tiếp cận khác nhau được đề xuất để lượng hóa tính chất này. Trong bối cảnh lý thuyết tập mờ truyền thống, các kỹ thuật định lượng thường dựa vào đặc điểm hình học của hàm thành viên — ví dụ như độ rộng của tập mờ, mức độ chồng lấp với các tập lân cận, hoặc diện tích dưới đường cong biểu diễn hàm.

Ngược lại, trong khuôn khổ đại số ngôn ngữ học (HA), một phương pháp tiếp cận mới được giới thiệu nhằm xác định tính mờ của từ ngôn ngữ dựa trên lập luận logic và ý nghĩa ngữ nghĩa. Theo [32], mức độ mờ của một từ ngôn

ngữ u phản ánh khả năng biến đổi ngữ nghĩa của x khi các phép biến đổi ngôn ngữ (gia tử) được áp dụng. Nói cách khác, từ ngôn ngữ nào càng dễ bị tác động bởi các gia tử như “very”, “more”, “slightly” thì mức độ mờ của nó càng cao, bởi ngữ nghĩa của từ không cố định mà dễ bị thay đổi bởi các ảnh hưởng từ bên ngoài.

Tập hợp các từ ngôn ngữ sinh ra từ một từ gốc x thông qua việc áp dụng các phép gia tử tạo nên tập H(u), phản ánh phổ biến đổi ngữ nghĩa của từ x. Do đó, H(u) có thể được sử dụng như một mô hình đại diện cho tính mờ của từ ngôn ngữ u. Độ lớn (số phần tử) của tập này được xem là một chỉ số đo lường định lượng cho tính mờ của u.

Định nghĩa 2.3 [32]: Với một HA tuyến tính AX = (X, G, H, ≤), ánh xạ fz: X → [0, 1] được gọi là độ đo tính mờ nếu thỏa mãn:

(i) fz là đầy đủ, nghĩa là fz(c-) + fz(c+) = 1 và ∑(h ∈ H) fz(hu) = fz(u), với mọi u ∈ X;

(ii) fz(u) = 0 nếu H(u) = {u}; và fz(0) = fz(W) = fz(1) = 0;

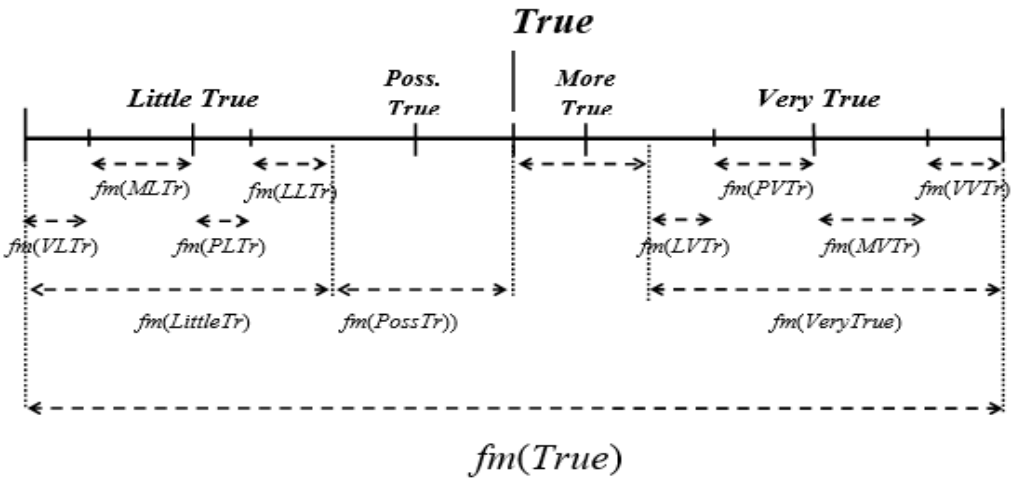
(iii) Với mọi u, v ∈ X và h ∈ H, ký hiệu μ(h) = fz(hu)/fz(u) = fz(hv)/fz(v), thì μ(h) là tỷ số không phụ thuộc vào u và v, gọi là độ đo tính mờ của gia tử h.

- Điều kiện (i) đảm bảo tính toàn vẹn của tập phần tử sinh G và tập gia tử H, cho phép mô hình hóa đầy đủ các giá trị ngữ nghĩa.

- Điều kiện (ii) đảm bảo mỗi từ ngôn ngữ được định danh rõ ràng về mặt ngữ nghĩa.

- Điều kiện (iii) giả định rằng mỗi gia tử có tác động thống nhất lên mọi từ ngôn ngữ trong tập X, không phụ thuộc vào ngữ cảnh cụ thể.

Hình 2.4 là minh họa trực quan thể hiện tập từ ngôn ngữ được sinh từ giá trị TRUTH bằng các phép gia tử, qua đó giúp hình dung mức độ biến đổi ngữ nghĩa và tính mờ tương ứng.



Hình 2.4. Độ đo tính mờ của biến Truth [32]

Một số tính chất của độ đo tính mờ đối với từ ngôn ngữ và các phép gia tử có thể được mô tả thông qua các mệnh đề hình thức, đóng vai trò thiết lập các quy tắc định lượng ngữ nghĩa trong khuôn khổ đại số gia tử. Những mệnh đề này thường phản ánh các nguyên lý sau:

Mệnh đề 1.1: [32] Với độ đo tính mờ fz và μ đã được định nghĩa trong Định nghĩa 2.3, ta có:

(i) fz(c−) + fz(c+) = 1 và ∑_{h ∈ H} fz(hu) = fz(u);

(ii) ∑_{j = -q}^{-1} μ(h_j) = α, ∑_{j = 1}^{p} μ(h_j) = β, với α, β > 0 và α + β = 1;

(iii) ∑_{x ∈ X_k} fz(x) = 1, trong đó X_k là tập các từ ngôn ngữ có độ dài đúng bằng k;

(iv) fz(hu) = μ(h) × fz(u), với ∀u ∈ X;

(v) Cho fz(c−), fz(c+) và μ(h) với ∀h ∈ H, khi đó với u = h_n ... h_1 c, trong đó c ∈ {c−, c+}, thì độ đo tính mờ của u được tính như sau:

fz(u) = μ(h_n) × ... × μ(h_1) × fz(c).

2.2.4. Định lượng ngữ nghĩa của từ ngôn ngữ

Theo phương pháp tiếp cận của lý thuyết tập mờ, mỗi tập mờ có thể được gán một giá trị định lượng duy nhất thông qua quá trình khử mờ (defuzzification), thường là giá trị đặc trưng như trọng tâm (centroid) của hàm thành viên tương ứng. Đây là cách tiếp cận phổ biến để chuyển đổi dữ liệu mờ sang giá trị số cụ thể trong các hệ thống điều khiển hoặc suy luận.

Tuy nhiên, trong lý thuyết đại số gia tử, các giá trị ngôn ngữ được sắp xếp theo một thứ tự ngữ nghĩa nội tại. Dựa trên cấu trúc này, HA xây dựng một hàm định lượng ngữ nghĩa (semantic quantification function) ánh xạ mỗi từ ngôn ngữ về một giá trị thực trong đoạn [0, 1]. Hàm này đảm bảo tính đơn điệu tăng, tức là nếu một từ có ý nghĩa "mạnh hơn" về mặt ngữ nghĩa thì sẽ được gán giá trị lớn hơn.

Nhờ vậy, HA cho phép ánh xạ định lượng giữa các giá trị ngôn ngữ và không gian số một cách nhất quán, hỗ trợ hiệu quả trong việc suy luận và tính toán với dữ liệu ngôn ngữ.

Định nghĩa 2.6: [32] Cho AX = ($\boldsymbol{X}$, $\boldsymbol{G}$, $\boldsymbol{H}$, $\leq$) là một HA tuyến tính. Ánh xạ Q_x: $\boldsymbol{X} \rightarrow$ [0,1] được gọi là một hàm định lượng ngữ nghĩa của AX nếu Q_x là ánh xạ 1-1 từ tập $\boldsymbol{X}$ vào đoạn [0, 1] và bảo toàn thứ tự trên $\boldsymbol{X}$, tức là $\forall u$, v $\in \boldsymbol{X}$, $u < v \Rightarrow Q_x(u) < Q_x(v)$ và $Q_x(\mathbf{0}) = 0$, $Q_x(\mathbf{1}) = 1$.

Dựa trên hai điều kiện cơ sở này, các tác giả trong [32] đã đề xuất một hàm định lượng ngữ nghĩa cho các từ ngôn ngữ ngôn ngữ được sinh ra từ HA. Phương pháp này cho phép ánh xạ mỗi từ ngôn ngữ về một giá trị số trong đoạn [0, 1], đồng thời đảm bảo giữ nguyên quan hệ thứ tự và cường độ ngữ nghĩa giữa các từ.

Trước khi đi đến hàm định lượng ngữ nghĩa, ta cần xét đến khái niệm "dấu của từ ngôn ngữ" – một thành phần cốt lõi trong việc thiết lập thứ tự và hướng tác động của các phép gia tử.

Định nghĩa 2.7: [32] Một hàm dấu Sg: X → {-1, 0, 1} là một ánh xạ được định nghĩa đệ qui như sau, trong đó h, h' ∈ H và c ∈ {c-, c+}:

(i1) Sg(c-) = -1, Sg(c+) = 1;

(i2) Sg(hc) = -Sg(c) nếu h âm đối với c; Sg(hc) = Sg(c) nếu h dương đối với c;

(i3) Sg(h'hu) = -Sg(hu), nếu h'hu ≠ hu và h' âm đối với h; Sg(h'hu) = Sg(hu), nếu h'hu ≠ hu và h' dương đối với h;

(i4) Sg(h'hu) = 0, nếu h'hu = hu.

Từ định nghĩa hàm dấu, chúng ta có tiêu chí để so sánh hu và u.

Mệnh đề 2.2. [32] Với bất kỳ h và u, nếu Sg(hu) = 1 thì hu > u; nếu Sg(hu) = -1 thì hu < u và nếu Sg(hu) = 0 thì hu = u.

Từ mệnh đề trên ta có:

- H(u) ≤ H(v), với bất kỳ u và v, tức là u ∉ H(u) và v ∉ H(v), đặc biệt là,

$$0 \le H(u) \le 1 \qquad (2.2)$$

- $Sgn(h_p u) = +1 \Rightarrow H(h_{-}qu) \le H(h_{-}q+1u) \le \ldots \le H(h_{-}1u) \le u \le H(h_1 u) \le H(h_2 u) \le \ldots \le H(h_p u)$ (2.3)

- $Sgn(h_p u) = -1 \Rightarrow H(h_{-}qu) \ge H(h_{-}q+1u) \ge \ldots \ge H(h_{-}1u) \ge x \ge H(h_1 u) \ge H(h_2 u) \ge \ldots \ge H(h_p u)$ (2.4)

Định nghĩa 2.8: [32] Cho fz là một độ đo tính mờ trên X của một đại số gia tử tuyến tính AX. Ánh xạ Qx: X → [0, 1] được xác định dựa trên độ đo tính mờ fz được định nghĩa bằng đệ qui như sau:

(i) $Qx(W) = \theta = fz(c-)$, $Qx(c-) = \theta - \alpha fz(c-) = \beta.fz(c-)$, $Qx(c+) = \theta + \alpha fz(c+)$;

(ii) $Qx(h_ju) = Qx(x) + Sign(h_ju)\{\sum[i=sign(j) \text{ to } j] \mu(h_i)fz(u) - \omega(h_ju)\mu(h_ju)fz(u)\}$, (2.5)

với mọi $j, -q \leq j \leq p$ và $j \neq 0$, trong đó: $\omega(h_ju) = 1/2[1 + Sg(h_ju)Sg(h_ph_ju)(\beta - \alpha)] \in \{\alpha, \beta\}$;

Với định nghĩa đã nêu, hàm định lượng ngữ nghĩa được chứng minh là thỏa mãn đầy đủ các tiêu chuẩn cần thiết của một ánh xạ định lượng trong khuôn khổ đại số gia tử. Cụ thể, hàm này đảm bảo:

- Tính hợp lệ về mặt hình thức, tức là nó là một hàm đơn ánh từ tập các từ ngôn ngữ **AX** vào đoạn [0, 1].

- Bảo toàn quan hệ thứ tự ngữ nghĩa giữa các từ ngôn ngữ: nếu một từ ngôn ngữ mang ngữ nghĩa mạnh hơn, nó sẽ được gán giá trị số lớn hơn.

- Tính trù mật (density) trong [0, 1], nghĩa là các giá trị định lượng của tập từ ngôn ngữ **AX** phân bố đều và liên tục trong toàn bộ miền định nghĩa, cho phép mô hình biểu diễn ngữ nghĩa một cách chi tiết và mượt mà.

Nhờ đó, hàm định lượng ngữ nghĩa không chỉ là công cụ định lượng đơn thuần, mà còn đóng vai trò nền tảng trong việc thiết lập cơ sở toán học để mô phỏng và suy luận trên không gian ngôn ngữ có tính mờ.

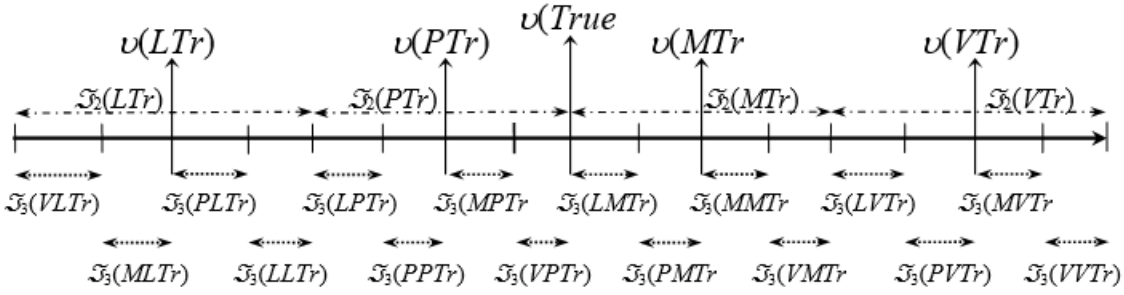
2.2.5. Khoảng tính mờ (fuzziness interval)

Khoảng tính mờ của một khái niệm mờ đóng vai trò then chốt trong việc phân tích và phát triển các mô hình ứng dụng trong các hệ thống xử lý ngôn ngữ mờ. Đây là khoảng giá trị định lượng nhằm biểu diễn phạm vi dao động ngữ nghĩa của một từ ngôn ngữ khi chịu tác động của các gia tử. Gọi $I([0, 1])$ là tập các đoạn con trên đoạn [0, 1], ký hiệu $|\bullet|$ là độ dài của đoạn "•".

Định nghĩa 2.9: [33] Khoảng tính mờ của các từ ngôn ngữ u ∈ X, ký hiệu Δfz(u), là một đoạn con của đoạn [0, 1], Δfz(u) ∈ I([0, 1]). Nếu nó có độ dài bằng độ đo tính mờ, |Δfz(u)| = fz(u), và được xác định bằng qui nạp theo độ dài của u như sau:

(i) Với độ dài của u bằng 1 (l(u) = 1), tức là u ∈ {c−, c+}, khi đó |Δfz(c−)| = fz(c−), |Δfz(c+)| = fz(c+) và Δfz(c−) ≤ Δfz(c+);

(ii) Giả sử u có độ dài n (l(u) = n) và khoảng tính mờ Δfz(u) đã được định nghĩa với |Δfz(u)| = fz(u). Khi đó tập các khoảng tính mờ {Δfz(hju): −q ≤ j ≤ p và j ≠ 0} ⊆ Itv([0,1]) được xây dựng sao cho nó là một phân hoạch của Δfz(u), và thỏa mãn |Δfz(hju)| = fz(hju) và có thứ tự tuyến tính tương ứng với thứ tự của tập {h−qu, h−q+1u, ..., hpu}, tức là nếu h−qu > h−q+1u > ... > hpu thì Δfz(h−qu) > Δfz(h−q+1u) > ... > Δfz(hpu) và ngược lại (xem hình 2.5). Dễ dàng thấy rằng hệ phân hoạch như vậy luôn tồn tại dựa vào tính chất i) trong Mệnh đề 2.1.



Hình 2.5. Khoảng tính mờ của các từ ngôn ngữ của biến TRUTH [33]

Một khoảng tính mờ được gọi là khoảng mờ mức *k* nếu độ dài của *u* bằng *k*, khi đó ta ký hiệu là $\mathfrak{I}_k(u)$ thay cho $\mathfrak{I}_{fz}(u)$.

2.2.6. Hệ khoảng mờ tương tự

Giả sử k là một số nguyên dương biểu thị độ dài tối đa cho các từ ngôn ngữ được xét. Một khía cạnh quan trọng khác trong ngữ nghĩa định lượng của các từ ngôn ngữ là khoảng mờ tương tự cấp độ k. Để xây dựng khái niệm này, ta xem xét tập hợp các từ ngôn ngữ có độ dài không vượt quá k, ký hiệu là:

$X(k) = \{ u \in X : |u| \leq k \}$

và một ánh xạ định lượng ngữ nghĩa Q_x đã được xác lập trước đó.

Mục tiêu là xây dựng một tập hợp các khoảng $\{ S^{(k)}(u) : u \in X(k) \}$ trong đoạn đơn vị [0,1], sao cho các khoảng này thỏa mãn hai điều kiện sau:

1. Tính bao hàm ngữ nghĩa: Mỗi giá trị định lượng $Q_x(u)$ phải nằm trong khoảng tương ứng $S^{(k)}(u)$, tức là:

$Q_x(u) \in S^{(k)}(u)$

Các giá trị thuộc khoảng này được xem là "tương tự" về ngữ nghĩa với $Q_x(u)$ ở mức độ trừu tượng k.

2. Tính phân hoạch: Tập các khoảng $\{ S^{(k)}(u) \}$ tạo thành một phân hoạch trên đoạn [0,1], nghĩa là:

- Các khoảng không giao nhau (hoặc chỉ giao tại biên),

- Hợp của tất cả các khoảng bằng đoạn [0,1].

Trường hợp cụ thể: $|H^-| = |H^+| = 1$

Giả sử tập các phép gia tử âm và dương chỉ chứa một phần tử, tức là:

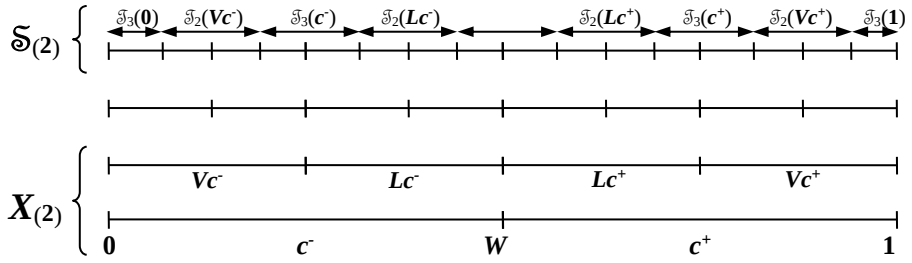
- $H^- = \{ L \}$ (ví dụ: “less”),

- $H^+ = \{ V \}$ (ví dụ: “very”).

Trong tình huống này, việc sinh ra các từ ngôn ngữ trong X(k) sẽ đơn giản hơn, vì mỗi từ chỉ có thể được tạo từ các tổ hợp lặp lại của các gia tử L và V áp dụng lên hai phần tử sinh c^- và c^+.

Để xây dựng các khoảng mờ tương tự mức k cho các từ ngôn ngữ thuộc X(k), ta sử dụng ngữ nghĩa topo — được hiểu là sự sắp xếp thứ tự và mối liên hệ ngữ nghĩa liên tục giữa các khoảng tính mờ ℑ_fz(u) tương ứng với các từ trong X(k+2). Mục tiêu là đảm bảo rằng các khoảng tương tự vừa phản ánh mối liên hệ ngữ nghĩa, vừa bao phủ không gian ngữ nghĩa một cách nhất quán và không chồng lắp tùy tiện.

Hệ khoảng tượng tự là một công cụ hữu dụng để phân hoạch miền tham chiếu của các biến, và được sử dụng trong các thuật toán sinh luật của các phương pháp tiếp cận dựa trên HA.



Hình 2.6. Minh họa hệ khoảng tương tự mức 2

2.3. HỆ MỜ DỰA TRÊN LUẬT

2.3.1. Các thành phần của hệ mờ

Một hệ thống mờ điển hình bao gồm bốn thành phần cơ bản như sau:

1. Cơ sở luật (Rule Base): Đây là tập hợp các luật mờ dạng "IF-THEN", phản ánh tri thức chuyên gia. Mỗi luật có dạng:

IF x_1 is A_1^k AND x_2 is A_2^k AND ... AND x_n is A_n^k THEN y is B^k

Trong đó:

- $x_1, x_2, ..., x_n$ là các biến đầu vào,

- y là biến đầu ra,

- A_i^k và B^k là các tập mờ tương ứng đại diện cho các khái niệm ngôn ngữ.

2. Cơ sở dữ liệu (Database): Chứa các hàm thành viên mô tả tập mờ. Những hàm này xác định ý nghĩa ngữ nghĩa của các thuật ngữ trong luật mờ. Cụ thể, mỗi tập mờ như A_i hay B^k được đặc trưng bởi một hàm $\mu: X \rightarrow [0, 1]$, với:

$\mu\_A(x)$ = mức độ mà x thuộc tập mờ A.

3. Bộ suy diễn mờ (Fuzzy Inference Engine): Thành phần này thực hiện quá trình suy luận dựa trên cơ sở luật và cơ sở dữ liệu. Nó áp dụng một kỹ thuật như Max-Min hoặc Max-Product để kết hợp các luật và xác định tập mờ kết quả cho biến đầu ra.

4. Bộ giải mờ (Defuzzifier): Có nhiệm vụ chuyển đổi kết quả mờ thành một giá trị cụ thể, dễ hiểu. Một số kỹ thuật giải mờ phổ biến gồm: phương pháp trọng tâm (centroid method), giá trị trung bình của cực đại (mean of maxima), và phương pháp giá trị lớn nhất (maximum membership).

Tóm lại, một hệ thống mờ vận hành dựa trên việc kết hợp giữa luật ngữ nghĩa, hàm thành viên, và cơ chế suy luận để xử lý thông tin không chắc chắn và đưa ra quyết định gần giống như con người.

2.3.2. Mục tiêu khi xây dựng FRBS

Trong quá trình thiết kế hệ mờ dựa trên luật, hai mục tiêu then chốt cần được cân bằng là: (i) độ chính xác thực thi và (ii) tính giải thích được (interpretability). Đây là hai mục tiêu mâu thuẫn, khi tăng cường một yếu tố thường làm giảm yếu tố còn lại.

Độ chính xác đã có thể định lượng bằng các chỉ số thống kê hoặc hàm mất mát. Tuy nhiên, tính giải nghĩa là một khái niệm phức tạp, chịu ảnh hưởng bởi nhiều yếu tố như số lượng luật, độ phủ và tính đơn giản của tập luật. Thậm chí, thuật ngữ dùng để mô tả đặc tính này hiện vẫn chưa được thống nhất trong cộng đồng nghiên cứu.

Phần sau sẽ trình bày các phương pháp tiêu biểu đã được đề xuất để đánh giá và cân đối hai mục tiêu này trong FRBS.

1) Đánh giá độ chính xác của FRBS

Đối với hệ mờ dựa trên luật, hiệu quả thực hiện có thể được đánh giá bằng các công thức toán học xác định rõ ràng. Trong bài toán phân lớp, độ chính xác được đo bằng tỷ lệ phần trăm số mẫu được phân loại đúngtrên tổng số mẫu, và càng cao thì hệ thống càng hiệu quả.

$$acc = \frac{N_{acc}}{N} * 100\% \longrightarrow max \quad (2.9)$$

trong đó N là số mẫu dữ liệu được phân lớp và N_{acc} là số mẫu dữ liệu được phân lớp chính xác.

2) Tính giải nghĩa được của FRBS

Tính giải nghĩa (interpretability) của FRBS là một khái niệm phức tạp và mang tính trừu tượng, phụ thuộc vào nhiều yếu tố khác nhau. Việc xác định một thước đo chuẩn hóa cho tính giải nghĩa hiện vẫn là chủ đề nghiên cứu mở. Một số công trình đã đề xuất phương pháp đánh giá tính giải nghĩa của hệ thống dựa trên luật mờ bằng cách phân loại các hệ thống này, đồng thời thiết lập một tập hợp các ràng buộc ở nhiều cấp độ. Hệ thống mờ dựa trên luật (FRBS) nào đáp ứng được càng nhiều ràng buộc thì được coi là có mức độ giải nghĩa càng cao. Theo Gacto et al. [10], hiện nay tồn tại hai hướng tiếp cận chính trong việc định nghĩa tính giải nghĩa:

1. Tính giải nghĩa được dựa trên độ phức tạp

Hướng tiếp cận này được phân thành hai mức:

+ Độ phức tạp ở mức cơ sở luật mờ (fuzzy rule base level): Thường sử dụng các độ đo sau:

- Số lượng luật: Hệ thống nên có càng ít luật càng tốt.
- Độ dài luật: Mỗi luật nên càng ngắn càng tốt (ít mệnh đề tiền đề).

+ Độ phức tạp ở mức phân hoạch mờ (fuzzy partition level): Thường sử dụng các độ đo sau:

- Số thuộc tính/biến: Việc sử dụng ít thuộc tính hoặc biến sẽ làm tăng tính giải nghĩa được của hệ luật.
- Số hàm thành viên: Số lượng hàm thành viên được sử dụng trong phân hoạch mờ không nên vượt quá 7±2 [25], theo nguyên tắc của Miller [25].

2. Tính giải nghĩa được dựa trên ngữ nghĩa

Hướng tiếp cận này cũng được chia thành hai mức:

+ Ngữ nghĩa ở mức phân hoạch mờ (mức từ - word level):

- Miền xác định của các biến phải được phủ hoàn toàn bởi các hàm thành viên của các tập mờ.
- Thuộc ít nhất một tập mờ: Tất cả các điểm dữ liệu phải thuộc vào ít nhất một tập mờ.
- Hàm thành viên chuẩn: Các hàm thành viên phải thuộc loại chuẩn (normal fuzzy sets), tức là mỗi hàm thành viên phải có ít nhất một điểm dữ liệu trong miền xác định của biến có độ thuộc bằng 1 (đỉnh của hàm).
- Phân biệt ngữ nghĩa: Các hàm thành viên biểu thị ngữ nghĩa của các tập mờ phải có khả năng phân biệt rõ ràng với nhau, tránh sự chồng lấn quá mức làm giảm tính rõ ràng của các khái niệm.

+ Ngữ nghĩa ở mức cơ sở luật (rule base level):

- Tính nhất quán: Cơ sở luật không chứa các luật mâu thuẫn (tức là các luật có cùng kết luận thì phải có cùng phần tiền đề).
- Số luật bị đốt cháy: Số lượng luật được kích hoạt (fired) bởi một mẫu dữ liệu đầu vào cụ thể càng ít càng tốt.

Khi xây dựng các thuật toán tiến hóa (evolutionary algorithms) để xây dựng các hệ mờ dựa trên luật từ dữ liệu, các hướng tiếp cận thường đặt mục tiêu đạt được các ràng buộc về tính giải nghĩa được bằng việc kết nhập các chỉ số liên quan để tạo ra một chỉ số tổng hợp. Và nó trở thành mục tiêu tính giải nghĩa được cần được tối ưu hóa của hệ luật trong quá trình tiến hóa. Dưới đây là một số cách kết hợp các chỉ số thể hiện tính giải nghĩa được của FRBS bao gồm:

i) Độ phức tạp của hệ luật

$$Comp = \sum_{m=1}^{M} length(R_m) \rightarrow min \quad (2.10)$$

ii) Trung bình độ dài của luật trong cơ sở luật

$$ave = \frac{\sum_{m=1}^{M} length(R_m)}{M} \rightarrow min \quad (1.11)$$

iii) Chỉ của Nauck [28]

$$I_{Nauck} = Comp \times Part \times Cov \quad (1.12)$$

Trong đó:

- Comp (Chỉ số độ phức tạp) được tính bằng tỷ lệ giữa số lớp đầu ra và tổng số điều kiện tiền đề của các luật trong FRBS. Chỉ số này phản ánh mức độ rút gọn hoặc chồng lắp giữa các luật.

- Part: là giá trị trung bình số tập mờ được gán cho mỗi biến đầu vào, biểu thị mức độ chi tiết của miền giá trị:

$$Part = \frac{\sum_{i=1}^{n} \frac{1}{T_j - 1}}{n} \quad (1.13)$$

với T_j là số tập mờ của phân hoạch của biến thứ j.

- *Cov (Coverage index)* là chỉ số biểu thị giá trị trung bình của cấp độ phủ (degree of coverage) trong toàn bộ phân hoạch mờ. Chỉ số này phản ánh mức độ mà các tập mờ chồng lấn và bao phủ không gian giá trị của các biến.

Tính giải nghĩa được của FRBS hiện vẫn là một vấn đề nghiên cứu còn bỏ ngỏ, với nhiều cách tiếp cận và tiêu chí đánh giá khác nhau chưa thống nhất. Do đó, vẫn cần phải tiếp tục nghiên cứu để cải thiện đặc tính giải thích của FRBS.

2.4. KẾT LUẬN CHƯƠNG 2

Chương 2 đề án đã tổng hợp các kiến thức nền tảng phục vụ cho quá trình nghiên cứu, bao gồm:

Lý thuyết tập mờ: trình bày các khái niệm cơ bản về tập mờ, biến ngôn ngữ, phương pháp xây dựng tập mờ, phân hoạch mờ và hệ mờ dựa trên luật.

Tổng quan về đại số gia tử (HA): đề cập đến các khái niệm cốt lõi như đại số gia tử tuyến tính, tuyến tính đầy đủ, phần tử sinh, gia tử, độ đo tính mờ, khoảng tính mờ và khoảng tương tự.

Hệ luật mờ: giới thiệu cấu trúc luật mờ, phương pháp lập luận trong bài toán phân lớp, cùng các chỉ số đánh giá hiệu quả hệ thống.

Những nội dung này đóng vai trò làm cơ sở cho việc xây dựng hệ mờ phát hiện bất thường trong mạng máy tính được trình bày ở Chương 3.

CHƯƠNG 3. XÂY DỰNG FRBS DỰA TRÊN HA PHÁT HIỆN BẤT THƯỜNG TRONG MẠNG MÁY TÍNH

Chương này trình bày quá trình thiết kế và triển khai mô hình phát hiện bất thường trong mạng máy tính, dựa trên hệ thống suy diễn mờ (FRBS) kết hợp với đại số gia tử (HA). Mở đầu chương là phần mô tả tổng quan kiến trúc của hệ thống, bao gồm các khối chức năng chính: tiền xử lý dữ liệu, phân hoạch mờ, xây dựng hệ luật, cơ chế suy diễn và đánh giá kết quả.

Tiếp theo, chương đi sâu vào từng bước trong quá trình xây dựng mô hình. Phần tiền xử lý tập trung vào việc xử lý dữ liệu thô, chuẩn hóa đầu vào và lựa chọn đặc trưng phù hợp từ tập dữ liệu KDD. Sau đó, các giá trị đầu vào được phân hoạch mờ bằng cách sử dụng đại số gia tử để xây dựng các tập ngôn ngữ có cấu trúc và ý nghĩa rõ ràng. Các luật mờ được sinh tự động từ tập dữ liệu huấn luyện bằng cách kết hợp các giá trị ngôn ngữ trong miền vào và miền ra, giúp mô hình phản ánh được mối quan hệ giữa các thông số mạng và hành vi bất thường.

3.1. GIẢI BÀI TOÁN PHÂN LỚP BẰNG FRBS VÀ HA

3.1.1. Tổng quan về giải bài toán phân lớp bằng FRBS

Phương pháp giải bài toán phân lớp bằng hệ mờ dựa trên luật là xây dựng một tập hợp ***S*** các luật mờ nhằm phân loại hoặc ánh xạ các mẫu dữ liệu thuộc tập *U* vào những giá trị nhãn lớp tương ứng trong tập ***C***. Mỗi luật trong hệ ***S*** có cấu trúc như biểu thức (3.1), trong đó vế phải của luật chính là một nhãn lớp thuộc tập ***C***

Trong bối cảnh bài toán phân lớp , mục tiêu là xây dựng một hệ thống dựa trên luật mờ, ký hiệu là ***S***. Hệ thống này thực hiện ánh xạ các mẫu dữ liệu từ không gian đầu vào ***U*** đến tập hợp các nhãn lớp ***C***. Các luật trong ***S*** được biểu diễn dưới dạng công thức (3.1), trong đó vế phải của mỗi luật là một nhãn lớp cụ thể từ tập ***C***. Điều này có nghĩa là mỗi luật mờ trong hệ thống sẽ liên kết một tập hợp các điều kiện mờ ở phần tiền đề (antecedent) với một dự đoán lớp duy nhất ở phần kết luận (consequent).

r_i: **If** X_1 is A_{i1} & … & X_n is A_{in} **then** X_{n+1} is C_i với i=1,.., N (3.1)

viết tắt là r_i: $A_i \Rightarrow C_i$.

Trong thiết kế hệ mờ dựa trên luật, hai mục tiêu quan trọng cần đạt được là: độ chính xác suy luận và tính giải thích được của hệ thống. Tuy nhiên, đây là hai mục tiêu thường mâu thuẫn với nhau: cải thiện độ chính xác có thể làm giảm tính giải nghĩa và ngược lại. Để giải quyết mâu thuẫn này, các phương pháp tối ưu hóa tiến hóa đa mục tiêu đã được đề xuất nhằm tự động sinh hệ luật FRBS, đồng thời đạt sự cân bằng (trade-off) giữa hai tiêu chí trên. Trong các nghiên cứu này, tính giải nghĩa thường được đặc trưng thông qua các thước đo độ phức tạp của hệ luật, chẳng hạn như số lượng luật, điều kiện tiền đề, hoặc cấu trúc phân hoạch mờ.

Quá trình phát triển các thuật toán giải bài toán phân lớp bằng FRBS gồm ba bước chính:

- Thiết kế phân hoạch mờ, nhằm xác định ngữ nghĩa tính toán của các thuật ngữ ngôn ngữ.
- Sinh tập luật mờ ứng cử, từ không gian các biến đầu vào và phân hoạch tương ứng.
- Tối ưu hóa hệ luật mờ $\boldsymbol{S}$ từ tập ứng cử, với mục tiêu:

+ Tối đa hóa độ chính xác $f_{acc}(\boldsymbol{S})$ – tỷ lệ phần trăm mẫu được phân lớp đúng , $f_{acc}(\boldsymbol{S}) \rightarrow max;$

+ Tối thiểu hóa độ phức tạp hệ thống thông qua:

- Số lượng luật $f_n(\boldsymbol{S})$

- Trung bình độ dài luật $f_a(\boldsymbol{S}) \rightarrow min$

- **Thiết kế phân hoạch mờ**

Trong nghiên cứu này, các phân hoạch mờ cho biến ngôn ngữ được xây dựng bằng cách sử dụng đại số gia tử để sinh các từ ngôn ngữ, sau đó xác định ngữ nghĩa tính toán của chúng thông qua các tập mờ tương ứng.

- **Bài toán sinh luật ứng cử**

Các đề xuất dựa trên lý thuyết tập mờ thường xây dựng luật bằng cách tổ hợp tất cả các giá trị ngôn ngữ của các biến đầu vào. Mỗi tổ hợp tương ứng với một luật mờ có dạng (3.1), như đã áp dụng trong các nghiên cứu của Cordón [6], Gacto [9], và Ishibuchi [14,15].

Trong một hệ dựa trên luật mờ, lớp kết luận (C_i) của một luật r_i được xác định dựa trên sự phân bố của các mẫu dữ liệu trong tập huấn luyện. Độ đốt cháy của luật r_q đối với một mẫu dữ liệu đầu vào p_j được tính toán như sau: : $\mu_{A_i}(p_j) = \prod_{k=1}^{n} \mu_{A_{ik}}(d_{ik})$, trong đó $\mu_{A_{ij}}(.)$ là hàm thành viên của tập mờ tương ứng với từ ngôn ngữ A_{ij}. Lớp kết luận C_i của luật r_i được xác định theo công thức (3.2) dưới đây:

$$C_i = \underset{C_h \in C}{argmax}\{conf(A_i \Rightarrow C_h)\} \qquad (3.2)$$

trong đó $conf = (A_i \Rightarrow C_h) = \frac{\sum_{p_k \epsilon C_h} \mu_{A_i(p_k)}}{\sum_{k=2}^{M} \mu_{A_j(p_k)}}$ là độ tin cậy của luật. Việc chọn lọc các luật mờ ứng cử được thực hiện dựa trên một tiêu chí đánh giá cụ thể. Trong các nghiên cứu của Ishibuchi [14, 15], luật được chọn dựa trên tích của độ hỗ trợ (coverage) và độ tin cậy (confidence) của luật. Tuy nhiên, nhược điểm chính của cách tiếp cận này là số lượng luật sinh ra tăng theo cấp số mũ khi số lượng thuộc tính trong tập dữ liệu tăng cao, gây khó khăn cho quá trình suy diễn và tối ưu hóa.

Với tập dữ liệu có *n* chiều, mỗi thuộc tính được phân hoạch bằng *T* tập mờ và t_{max} là chiều dài tối đa của luật khi đó không gian luật phải xem xét là $T^n * \sum_{t=1}^{t_{max}\Sigma} C_n^t$.

Trong [3, 4, 33] Nguyễn Cát Hồ và Dương Thăng Long đề xuất phương pháp sinh luật dựa trên mẫu dữ liệu. Các tác giả sử dụng đại số gia tử để phân hoạch các miền tham chiếu của các biến và xây hệ khoảng mờ tương tự trên các biến. Từ đó xây dựng các siêu hộp. Để sinh luật, thuật toán duyệt qua từng mẫu dữ liệu, mỗi mẫu dữ liệu sẽ rơi vào 1 siêu hộp. Từ đó sinh ra một luật có độ dài *n*, sau dos sinh ra các luật có độ dài $l \le n$ bằng cách thay thế ngẫu nhiên một số tiền đề bằng Don'tcare . Khi đó không gian luật tối đa phải xem xét theo phương pháp sinh luật này là $N * \sum_{t=1}^{t_{max}} C_n^t$, phương pháp sinh luật này đã làm làm giảm không gian luật phải xem xét đi nhiều so với phương pháp sinh luật bằng tổ hợp, mặc dù vậy không gian luật phải xem xét vẫn còn rất lớn so với phương pháp sinh luật của Mansoori trong [19].

Trong [19] Mansoori đã đề xuất một giải thuật di truyền sinh hệ luật ứng cử gọi là SGERD. Việc chọn ra hệ luật tối ưu từ hệ luật ứng cử dựa trên trên độ thích nghi của luật. Ưu điểm của thuật toán này có là số lượng các luật ứng cử sinh ra rất ít, và thuật giải di truyền có số thế hệ hữu hạn. Tuy nhiên hệ luật tối ưu cuối cùng chỉ lựa chọn theo kinh nghiệm, hơn nữa các từ ngôn ngữ sử dụng trong hệ luật chỉ là các nhãn gắn với các tập mờ tam giác cố định và không được điều chỉnh thích nghi với từng tập dữ liệu. Đó là một trong các nguyên nhân quan trọng làm cho các FRBS được sinh ra bằng thuật toán SGERD có hiệu quả phân lớp không cao.

Tiếp cận dựa trên đại số gia tử, được đề xuất bởi Nguyễn Cát Hồ [4, 33] và Dương Thăng Long [3], cho phép sinh luật từ từng mẫu dữ liệu: mỗi mẫu tạo ra một luật độ dài *n*, từ đó sinh các luật ngắn hơn với độ dài $t \le n$. Số lượng luật ứng cử cần xét theo phương pháp này là $N * \sum_{t=1}^{t_{max}} C_n^t$. Dù nhỏ hơn đáng kể so với phương pháp tổ hợp đầy đủ, số lượng luật vẫn còn rất lớn so với hướng tiếp cận của Mansoori [19].

Để giảm số lượng luật cần xét, Mansoori đề xuất thuật toán di truyền SGERD, sinh hệ luật ứng cử với số lượng rất ít và số thế hệ hữu hạn. Tuy nhiên, việc chọn luật tối ưu chỉ dựa trên đánh giá heuristic và sử dụng các nhãn ngôn ngữ cố định (tập mờ tam giác không điều chỉnh theo dữ liệu), dẫn đến hiệu quả phân lớp của FRBS sinh ra không cao.

- **Vấn đề tối ưu hóa hệ luật mờ**

Để tối ưu hóa hệ thống FRBS, hầu hết các phương pháp hiện nay đều sử dụng giải thuật di truyền, như trong các nghiên cứu của Dương Thăng Long [3], Cordón [6], Gacto [9], Ishibuchi [14,15], và Nguyễn Cát Hồ [4,33].

Đề án này áp dụng thuật toán OPHA-SGERD được đề xuất trong [4]. Trong đó, tập luật ứng cử được sinh bằng thuật toán HA-SGERD, phát triển từ SGERD của Mansoori [19], nhưng sử dụng ngữ nghĩa tính toán dựa trên đại số gia tử (HA). Thuật toán khắc phục được một số hạn chế của các phương pháp trước đó, cụ thể:

- Không gian tìm kiếm tham số được rút gọn, nhờ tối ưu ngữ nghĩa từ ngôn ngữ qua HA.

- Số lượng luật ứng cử giảm đáng kể so với phương pháp tổ hợp toàn bộ không gian ngữ nghĩa.

Cấu trúc thuật toán gồm hai pha chính:

- **Pha 1** – Thiết kế phân hoạch mờ: bằng cách đi tìm bộ tham số tính mờ tối ưu hóa của HA sử dụng thuật toán di truyền OP-PARHA.

- **Pha 2** – Tìm kiếm hệ luật tối ưu: Dựa trên bộ tham số từ pha 1, áp dụng thuật toán HA-SGERD để sinh tập luật ứng cử **R**; sau đó, thuật toán di truyền HA-OFRB được sử dụng để chọn hệ luật tối ưu.

3.1.2 Thuật toán OPHA-SGERD

1) Thiết kế phân hoạch mờ (ngữ nghĩa tính toán của từ)

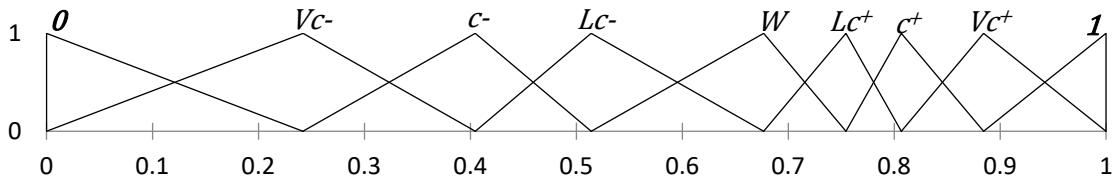
Phân hạch mờ cảu các biến X_j được xây dựng từ các tập mờ của các từ ngôn ngữ A_{ji} ($i=1,..,|X_j|$), được sinh ra từ đại số gia tử tuyến tính HA, AX_j. Hàm thuộc của các từ được biểu diễn bằng các tập mờ tam giác (triangular fuzzy sets), được tổ chức thành một phân hoạch dạng đơn thể hạt.

Chúng ta giả định rằng mỗi HA AX_j có hai phần tử sinh, một gia tử dương V_j (Very), có tác dụng tăng cường ngữ nghĩa. Một gia tử âm L_j (Little), có tác dụng làm giảm ngữ nghĩa.

Tập mờ của A_{ji} được xác định như sau:

- Giá trị định lượng $Q_x(A_{ji})$ của từ A_{ji} là lõi (core) của tập mờ;
- Giá trị định lượng $Q_x(A_{j(i-1)})$ của từ liền kề bên trái A_{ji} là điểm chân bên trái.
- Giá trị định lượng $Q_x(A_{j(i+1)})$ của từ liền kề bên trái A_{ji} là điểm chân bên phải.

Từ hằng **0**, điểm bên trái nhất của độ hỗ trợ sẽ trùng với lõi của nó. Từ hằng **1**, điểm bên phải nhất của độ hỗ trợ sẽ trùng với lõi của nó. Hình 3.1 minh họa một ví dụ về phân hoạch mờ được xây dựng theo phương pháp này, với các từ có độ dài không quá 2.



Hình 3.1. Một phân hoạch mờ đơn thể hạt được xây dựng dựa trên HA

2) Tiêu chuẩn chọn luật

Để tối ưu hóa quá trình tìm kiếm và chỉ giữ lại các luật có chất lượng cao trong một hệ luật, các thuật toán cần áp dụng một tiêu chuẩn chọn luật trong quá trình sinh tập luật ứng cử. Giả sử $f\ \ (A_i \Rightarrow C_i)$ là độ đo của một tiêu chuẩn

chọn luật cho một luật i cụ thể. $f\ (A_i \Rightarrow C_i)$ có thể được định nghĩa theo một số cách, trong đó một phương pháp phổ biến và hiệu quả là:

Dựa trên hiệu số độ đốt cháy: Tiêu chuẩn này xác định $f\ (A_i \Rightarrow C_i)$ bằng hiệu giữa tổng độ đốt cháy của luật đối với các mẫu dữ liệu được phân loại đúng và tổng độ đốt cháy của luật đối với các mẫu dữ liệu bị phân loại sai [19]. Đây là một tiêu chuẩn được sử dụng rộng rãi và chứng minh hiệu quả trong nhiều nghiên cứu.

$$f(A_i \Rightarrow C_i) = \sum_{p_k \in Class C_i} \mu_{A_i}(p_k) - \sum_{p_k \notin Class C_i} \mu_{A_i}(p_k) \qquad (3.3)$$

- Mansoori et al. [19] đã chỉ ra rằng mỗi luật mờ luôn xác định hai không gian quan trọng: không gian phủ (covering space) và không gian quyết định (decision space). Các mẫu dữ liệu mà luật đó được đốt cháy được gọi là không gian phủ của luật. Các mẫu dữ liệu được phân lớp chính xác được gọi là không gian quyết định của luật.

Do các luật gần nhau không gian phủ của chúng có thể giao nhau, việc xác định chính xác không gian quyết định của một luật trong quá trình sinh luật là một thách thức. Để vượt qua vấn đề này, Mansoori et al. [19] đề xuất sử dụng một ngưỡng (τ_i) cho mỗi luật r_i. Theo đó, các mẫu dữ liệu có độ đốt cháy lớn hơn ngưỡng (τ_i) sẽ được ước lượng là thuộc về không gian quyết định của r_i. Từ đó, [19] đã đề xuất tiêu chuẩn chọn luật sau:

$$\boldsymbol{f(A_i \Rightarrow C_i) = \sum_{p_k \in Class C_i} \mu_{A_i}(p_k) - \sum_{p_k \notin Class C_i} \mu_{A_i}(p_k) + \gamma_i(1 - \tau_i)} \quad (3.4)$$

trong đó γi là số mẫu dữ liệu có độ đốt cháy luật r_i cao hơn τ_i. τ_i là tham số ngưỡng được xác định như sau $\tau_i = 0.5^{l_i}$ với l_i là độ dài của luật r_i.

Trong nghiên cứu của [4], các tác giả chỉ ra rằng việc sử dụng một giá trị ngưỡng cố định để xác định liệu một mẫu dữ liệu có thuộc về không gian quyết định của một luật hay không là chưa tối ưu. Lý do là mật độ (density) và phân bố (distribution) của các mẫu dữ liệu khác nhau đáng kể giữa các tập dữ liệu.

Điều này hàm ý rằng, tham số ngưỡng cần phải phụ thuộc vào từng tập dữ liệu cụ thể để phản ánh chính xác đặc điểm của dữ liệu.

Do đó, [4] đã đề xuất một tiêu chuẩn chọn luật cải tiến (Công thức 3.5), phát triển từ tiêu chuẩn (3.4). Trong tiêu chuẩn mới này, giá trị xác định ngưỡng là một hàm phụ thuộc vào tham số ρ, với $\rho \in (0,1)$. Giá trị của r được xác định tùy chỉnh cho từng tập dữ liệu, đồng thời với quá trình tìm kiếm tham số mờ tối ưu của HA. Cách tiếp cận này giúp tiêu chuẩn chọn luật trở nên linh hoạt và hiệu quả hơn trong việc xử lý các tập dữ liệu đa dạng.

$$f(A_i \Rightarrow C_i) = \sum_{p_k \in ClassC_i} \mu_{A_i}(p_k) - \sum_{p_k \notin ClassC_i} \mu_{A_i}(p_k) + \gamma_i(1 - \rho^{l_i}) \quad (3.5)$$

Trong (3.5), γ_i là số mẫu dữ liệu có độ đốt cháy luật r_i lớn hơn ρ^{l_i}.

3) Thuật toán sinh hệ luật ứng cử

Trong nghiên cứu [4], các tác giả đã phát triển một thuật toán mang tên HA-SGERD. Thuật toán này được thiết kế để sinh các luật ứng cử (candidate rules) bằng cách tích hợp thuật toán SGERD của Mansoori [19]. Điểm đặc biệt của HA-SGERD là tập từ ngôn ngữ được sử dụng cho mỗi biến được tạo ra bởi HA, và ngữ nghĩa tính toán của mỗi từ được xác định dựa trên ngữ nghĩa vốn có của chúng.

Mỗi cá thể của quần thể là một luật mờ *r* có *n* điều kiện tiền đề và lớp kết luận trong *C*, r_q: $A_q \rightarrow C_q$, trong đó $A_q = (A_q[1], \ldots, A_q[n])$, $A_q[i]$ là điều kiện tiền đề thứ *i* của luật và nhận giá trị trong tập từ ngôn ngữ $X_{j(k_j)}$ được sinh ra từ HA AX_j và $\{Don't care\}$ với $\mu_{\text{Don'tcare}}(x) = 1$. Ta gọi các biến ứng với điều kiện tiền đề có giá trị *Don't care* là các biến *không được kích hoạt*, các biến ứng với điều kiện tiền đề có giá trị thuộc $X_{j(k_j)}$ là các biến *được kích hoạt* của luật.

Trong các phương pháp tối ưu hóa dựa trên quần thể, mỗi cá thể đại diện cho

một luật mờ r_i. Luật này có *n* điều kiện tiền đề và một lớp kết luận từ tập C, được biểu diễn dưới dạng r_i: $A_i \rightarrow C_i$.

Trong mô hình này, mỗi luật được biểu diễn thông qua một vector điều kiện tiền đề, ký hiệu là $A_i = (A_i[1], A_i[2], ..., A_i[n])$. Mỗi phần tử $A_i[j]$ thể hiện điều kiện tiền đề thứ j tương ứng với biến đầu vào X_j. Giá trị của $A_i[j]$ có thể thuộc tập từ ngôn ngữ $X_j(k_j)$ – được sinh bởi hàm định lượng HA tương ứng với biến X_j – hoặc có thể nhận giá trị đặc biệt là "Don't care".

Dựa trên giá trị của các điều kiện tiền đề, các biến trong luật được phân thành hai loại:

- Biến không kích hoạt: Là các biến có điều kiện tiền đề là "Don't care", tức không ảnh hưởng đến việc luật đó có được kích hoạt hay không.

- Biến kích hoạt: Là các biến có điều kiện nằm trong tập từ ngôn ngữ $X_j(k_j)$, đóng vai trò chính trong việc xác định sự phù hợp của luật với mẫu dữ liệu.

Trong mỗi thế hệ tiến hóa, một tập tối đa Q luật có độ đo thích nghi cao nhất được lựa chọn từ tập luật ứng cử, cho từng lớp, để tạo thành quần thể hiện tại. Độ đo thích nghi này được xác định theo một tiêu chí đánh giá đã được xác lập từ trước.

Từ quần thể hiện tại, quá trình tạo luật con được thực hiện. Mỗi luật con có độ dài lớn hơn cha mẹ một đơn vị – tức là được mở rộng thêm một điều kiện tiền đề. Những luật con nào có độ đo thích nghi cao hơn luật gốc sẽ được giữ lại cùng với toàn bộ quần thể hiện tại, để hình thành tập luật ứng cử cho vòng tiến hóa tiếp theo.

Quần thể ban đầu được hình thành từ tất cả các luật có chiều dài bằng một, bằng cách liệt kê mọi kết hợp có thể giữa các giá trị ngôn ngữ trong $X_j(k_j)$ với

j = 1,…,n. Cách khởi tạo này đảm bảo tính đa dạng cần thiết cho quá trình tiến hóa luật mờ..

Một số ký hiệu: $|R(C_q)|, |\mathcal{R}(C_q)|$ lần lượt là số luật của tập luật $R(C_q)$ và $\mathcal{R}(C_q)$; $R(C_q)$, $\mathcal{R}(C_q)$, $R'(C_q)$ là các tập luật có cùng lớp kết luận C_q; $\mathcal{R}$ là tập luật ứng cử; R là tập các luật của quần thể hiện tại, R' là tập luật bổ trợ cho quá trình tạo sinh; $r_q.fitness$ và $r_q.Class$ là độ đo thích nghi và lớp kết luận của luật r_q.Trong bài toán phát hiện bất thường trong mạng $r_q.Class$ nhận một trong 2 giá trị là 0 hoặc 1, 0 – là bình thường, 1 - là bất thường.

Hàm $Sort(\mathcal{R}(C_q))$ thực hiện sắp xếp các luật trong tập luật $\mathcal{R}(C_q)$ theo độ đo thích nghi của luật giảm dần.

Algorithm HA-SGERD(D, m, n, Q, l_{max}, X, f)

Input:

- Tập mẫu dữ liệu $D = \{(p_i, C_i): i = 1,..,N\}$, số lớp m, số thuộc tính n
- Số luật Q được chọn ứng với mỗi lớp, tập các tập từ ngôn ngữ $X = \{X_{j(k_j)} : j = 1,..,n\}$
- Độ dài tối đa của luật: l_{max}
- Độ đo thích nghi f của luật (sử dụng 1 trong 3 tiêu chuẩn (3.3), (3.4) hoặc (3.5))

Output:

- Tập luật mờ $\mathcal{R}$ có tối đa $m*Q$ luật

Method: Phương pháp tiến hóa sinh hệ luật

Begin

1. Khởi tạo $\mathcal{R}$ bằng rỗng;
2. **for** $j \leftarrow 1$ **to** n **do**
   for each x **in** $X_{j(k_j)}$ **do**
3. Sinh luật r_q có $A_q[l]$ = *Don'tcare* với $l = 1,..,n$ *nếu* $l \neq j$ và $A_q[j] = x$
4. *nếu* $l = j$;
5. $r_q.Class \leftarrow \underset{C_h \in C}{\boldsymbol{argmax}}\{\boldsymbol{conf}(\boldsymbol{A_q \Rightarrow C_h})\}$; *//xác định lớp kết luận*
6. $r_q.fitness \leftarrow f(r_q)$; *//tính độ đo thích nghi của luật r*
7. $\mathcal{R} \leftarrow \mathcal{R} \cup \{r_q\}$;
8. **endfor**
9. **endfor**
10. $g \leftarrow 1$;
11. Khởi tạo *R'* bằng tập luật rỗng; *//R' là tập bổ trợ sử dụng trong quá trình*
12. *tạo sinh*
13. **repeat**
14. **for each** C_q **in** C **do**
15. Khởi tạo $\mathcal{R}(C_q)$ bằng tập luật rỗng;*//*$\mathcal{R}(C_q)$ *chứa các luật trong* $\mathcal{R}$
16. *có vế phải là* C_q;
17. **for each** r **in** $\mathcal{R}$ **do**
18. **if** $r.Class = C_q$ **then** $\mathcal{R}(C_q) \leftarrow \mathcal{R}(C_q) \cup \{r\}$;
19. **endfor**
20. *Sort*($\mathcal{R}(C_q)$); *//sắp xếp tập luật* $\mathcal{R}(C_q)$ *theo độ đo thích nghi của*
21. *luật giảm dần*
22. **endfor**
23. Khởi tạo R bằng tập luật rỗng;
24. **for each** C_q ***in*** C **do**
25. **if** $g > 1$ **then**
26. Đặt $R(C_q)$ gồm $min\{Q, |\mathcal{R}(C_q)|\}$ luật đầu tiên của $\mathcal{R}(C_q)$;
27. **else**

28. $Q \leftarrow min\ \{Q, |\mathcal{R}(C_q)|/2\}$
29. Đặt $R(C_q)$ gồm Q luật đầu tiên trong $\mathcal{R}(C_q)$;
30. $\mathcal{R}(C_q) \leftarrow \mathcal{R}(C_q) \backslash R(C_q)$;// Loại các luật có trong $R(C_q)$ ra khỏi
31. $\mathcal{R}(C_q)$
32. Đặt $R'(C_q)$ gồm Q luật đầu tiên trong $\mathcal{R}(C_q)$; //*xác định tập*
33. *bổ trợ R' của lớp* C_q
34. $R' \leftarrow R' \cup R'(C_q)$;
35. **endif**
36. $R \leftarrow R \cup R(C_q)$;

endfor

$g \leftarrow g+1$;

if $g < l_{max}$ **then** $\mathcal{R} \leftarrow$ *Reproduce*(*R, R', D, m, n, X, f*)**;** *//thực hiện thuật toán tạo sinh*

until ($g > l_{max}$) **or** ($\mathcal{R}$ trùng R);

return $\mathcal{R}$;

End

Quá trình tạo sinh luật cho thế hệ kế tiếp được thực hiện thông qua thuật toán Reproduce. Thuật toán này sử dụng các luật từ quần thể hiện tại R và quần thể bổ trợ R' để tạo ra một tập hợp các luật ứng cử (candidate rules). Tập này bao gồm tất cả các luật từ R và các luật con mới có độ đo thích nghi cao hơn luật cha mẹ đã sinh ra chúng.

Đối với mỗi luật r_q có lớp kết luận C_q trong quần thể R, quá trình tạo sinh diễn ra như sau:

Chọn luật cha thứ hai: Ngẫu nhiên chọn một luật r_p từ tập hợp các luật trong R có cùng lớp kết luận C_q.

Xử lý trùng lặp: Nếu r_p trùng với r_q, chọn lại r_p một cách ngẫu nhiên từ các luật có cùng lớp kết luận C_q trong quần thể bổ trợ R'.

Chọn biến để tạo sinh: Ngẫu nhiên chọn một biến X_j từ các biến được kích hoạt của luật r_p.

Kiểm tra và tạo luật con: Nếu biến X_j tương ứng trong luật r_q có giá trị là "Don't care", thuật toán sẽ tạo ra các luật con của rq bằng cách thay thế giá trị "Don't care" đó bằng một giá trị ngôn ngữ bất kỳ từ tập $X_{j(kj)}$.

Lớp kết luận của các luật con này được xác định theo công thức (3.2), và độ thích nghi của chúng được tính theo tiêu chuẩn chọn luật đã nêu trong mục 2. Chỉ những luật con nào có độ đo thích nghi cao hơn luật r_q mới được giữ lại để làm luật ứng cử cho thế hệ kế tiếp. Nếu biến X_j tương ứng trong luật rq có giá trị khác "Don't care", sẽ không thực hiện tạo sinh trên luật r_q.

Các hàm hỗ trợ được sử dụng là: Random(R, C_p): Chọn ngẫu nhiên một luật từ tập luật R có cùng lớp kết luận C_p. Randomactive(r_p): Chọn ngẫu nhiên chỉ số của một biến từ các biến được kích hoạt của luật r_p.

Algorithm *Reproduce*(*R, R', D, m, n, X, f*)

***Input*:**

- Tập luật mờ *R,* tập luật mờ bổ trợ *R'*, tập mẫu dữ liệu $D = \{(p_i, C_i): i = 1,..,N\}$
- Số lớp *m,* số thuộc tính *n*, tập các tập từ ngôn ngữ $X = \{X_{j(k_j)} : j = 1,..,n\}$
- Độ đo thích nghi *f* của luật (sử dụng 1 trong 3 tiêu chuẩn (3.3), (3.4) hoặc (3.5))

***Output**:*

- Tập luật mờ $\mathcal{R}$

Method: Tạo sinh hệ luật ứng cử cho quần thể kế tiếp

Begin

1. $\mathcal{R} \leftarrow R$;
2. **for each** r_q **in** R **do**
3. $r_p \leftarrow Random(R, C_q)$;

4. **if** $r_p \equiv r_q$ **then** $r_p \leftarrow RAndom(R', C_q)$;
5. $j \leftarrow Randomactive(r_p)$;
6. **if** $A_q[j] = Don'tcare$ **then** //A_q là vế trái của luật r_q
7. **for each** x **in** $X_{j(k_j)}$ **do**
8. Sinh luật r với $A[l] \leftarrow A_q[l]$ với $l = 1,..,n$;
9. $A[j] \leftarrow x$; //A là vế trái của luật r
10. $r.Class \leftarrow \underset{C_h \in C}{\boldsymbol{argmax}}\{\boldsymbol{conf}(\boldsymbol{A_r} \Rightarrow \boldsymbol{C_h})\}$
$r.fitness \leftarrow f(r)$; //tính độ đo thích nghi của luật
11. **if** $r.fitness > r_q.fitness$ **then** $\mathcal{R} \leftarrow \mathcal{R} \cup \{r\}$;
12. **endfor**
13. **endif**
14. **endfor**
15. **return** $\mathcal{R}$;
16.

End

4) Thuật toán OP-PARHA thiết kế ngôn ngữ [4]

Để cải thiện khả năng phân loại của hệ thống luật mờ trong nghiên cứu, việc tinh chỉnh các tập mờ trong Hệ luật Ngôn ngữ (Linguistic Rule-Based System) cho phù hợp với từng bộ dữ liệu cụ thể là một yếu tố then chốt. Trong khuôn khổ này, các tập mờ được xây dựng trên cơ sở ngữ nghĩa nội tại của các biểu thức ngôn ngữ, thông qua việc định lượng hóa bằng hàm định lượng ngữ nghĩa HA (Hàm HA – Heuristic Approximation).

Thay vì xây dựng thủ công các hàm thành viên cho từng trường hợp, cách tiếp cận này cho phép chuẩn hóa biểu diễn mờ dựa trên một khung ngữ nghĩa nhất quán. Do đó, nhiệm vụ điều chỉnh hệ thống mờ chủ yếu quy về việc tối ưu các tham số mờ trong hàm định nghĩa HA, từ đó nâng cao khả năng thích ứng của hệ thống với dữ liệu huấn luyện cụ thể.

Giả sử mỗi biến đầu vào X_j được gán với hai khái niệm ngôn ngữ đối lập: một giá trị tích cực (ký hiệu là Vj) và một giá trị tiêu cực (ký hiệu là Lj), được áp dụng thống nhất trên toàn bộ các biến đầu vào. Khi đó, bài toán hiệu chỉnh hệ thống mờ sẽ tương đương với việc tìm kiếm một bộ tham số tối ưu — được gọi là vector tham số op — để điều khiển sự phân bố của các tập mờ do hàm HA xác định :

- Nếu sử dụng tiêu chuẩn chọn luật (3.5), $op = \{(o\mu^j_{fmc-}, o\mu^j_L, ok_j, o\rho): j = 1,..,n\}$.
- Nếu sử dụng các tiêu chuẩn chọn luật (3.3), (3.4), $op = \{(o\mu^j_{fmc-}, o\mu^j_L, ok_j) : j=1,..,n\}$.

Để tìm bộ tham số mờ tối ưu cho HA, chúng tôi đã thiết kế một thuật giải di truyền (Genetic Algorithm) với sơ đồ mã hóa nhị phân. Hàm mục tiêu (objective function), ký hiệu là *perf*(R,D), đánh giá hiệu quả phân lớp của hệ luật *R* trên toàn bộ tập mẫu dữ liệu D (sử dụng phương pháp "Test All"). Hệ luật *R* được sinh ra từ thuật toán HA-SGERD. Các toán tử đột biến (mutation), lai ghép (crossover) và lựa chọn quần thể (population selection) cho thế hệ kế tiếp được thừa kế từ [2]. Thuật toán chi tiết được trình bày trong [4].

Algorithm OP-PARHA(*D, m, n, Q, f, PAR, Npop, Gmax, Pmu, Pcross, lchrom*)

Input:

- Tập mẫu dữ liệu $D = \{(p_i, C_i): i = 1..N\}$, số lớp kết luận *m*, số thuộc tính *n*
- Số luật *Q* được chọn tối đa trên mỗi lớp
- Độ đo thích nghi *f* của luật (sử dụng 1 trong 3 tiêu chuẩn (3.3), (3.4) hoặc (3.5))
- Tập các ràng buộc tham số tính mờ của HA $PAR = \{(min\,\mu^J_{fmc-}, max\mu^J_{fmc-}, min\,\mu^J_I, max\mu^J_I, mink_j, maxk_j): j = 1,..,n\}$, với tiêu chuẩn (3.5) thêm ràng buộc $\{min\rho, max\rho\}$
- *Npop* là số cá thể trong quần thể, *Gmax* là số thế hệ tiến hóa, *Pmu* là xác suất đột biến, *Pcross* là xác suất lai ghép, *lchrom* là chiều dài gen

Output:

- Bộ tham số tối ưu $op = \{(o\mu^J_{fmc-}, o\mu^J_I, ok_j, o\rho): j = 1,..,n\}$ với tiêu chuẩn chọn luật (3.5) hoặc $op = \{(o\mu^J_{fmc-}, o\mu^J_I, ok_j): j = 1,..,n\}$ với các tiêu chuẩn còn lại.

Method: Phương pháp tối ưu tham số tính mờ của HA

Begin

1. $i \leftarrow 0$;
2. Khởi tạo quần thể $POP_0 = \{p_{0,1}, \ldots, p_{0,Npop}\}$;
3. *op.objvalue* $\leftarrow$ 0; *//khởi tạo giá trị hàm mục tiêu của biến lưu bộ tham*

4. *số tối ưu*
5. **repeat**
6. **for each** *p* **in** POP_i **do**
7. Xây dựng tập các tập từ ngôn ngữ $X = \{X_{j(k_j)} : j = 1,..,n\}$ bằng HA với bộ tham số tính mờ của *p*;
8. $R \leftarrow$ HA-SGERD(*D, m, n, Q, X, f*) ; *//tạo hệ luật R với tập các*
9. *tập từ ngôn ngữ X*
10. *p.objvalue* $\leftarrow$ *perf(R, D) ; //tính giá trị hàm mục tiêu của cá thể*
11. *p*
12. **if** *op.objvalue* < *p.objvalue* **then** *op* $\leftarrow$ *p*;
13. **endfor**
14. $i \leftarrow i+1$;
15 **if** $i \leq Gmax$ **then**
16. Thực hiện lai ghép, đột biến;
17. Lựa chọn quần thể cho thế hệ kế tiếp: POP_i;
endif

until $i > Gmax$;
return *op; //trả lại bộ tham số tối ưu*
End

5) Thuật toán HA-OFRB tối ưu hóa hệ luật

Sau khi xác định được bộ tham số tối ưu thông qua thuật toán OP-PARHA, chúng ta sử dụng bộ tham số này trong thuật toán HA-SGERD để sinh ra $m \times Q$ luật ứng cử, ký hiệu là *R*. Từ tập luật ứng cử R, thách thức tiếp theo là tìm kiếm một hệ luật con tối ưu *S* nhằm đạt được đồng thời các mục tiêu sau:

- Tối đa hóa $f_p(S) \rightarrow$ max
- Tối thiểu hóa $f_n(S) \rightarrow$min
- Tối thiểu hóa $f_a(S) \rightarrow$min

Đây là một bài toán tối ưu hóa đa mục tiêu, để có thể áp dụng hiệu quả thuật toán di truyền, chúng ta cần chuyển đổi bài toán này thành một bài toán đơn mục tiêu bằng cách kết hợp các mục tiêu riêng lẻ.

Để giải quyết vấn đề này, trong [4], các tác giả đã thiết kế thuật toán HA-OFRB. Thuật toán này dựa trên một giải thuật di truyền với sơ đồ mã hóa nhị phân. Hàm mục tiêu *f*(*S*) của HA-OFRB được kết hợp từ ba mục tiêu đã nêu trên, sử dụng phương pháp đề xuất bởi Ishibuchi và cộng sự. trong [15].

$$f(\boldsymbol{S}) = w_p.f_p(\boldsymbol{S}) + w_n.f_n(\boldsymbol{S})^{-1} + w_a.f_a(\boldsymbol{S})^{-1} \rightarrow max$$

trong đó $0 < w_p, w_n, w_a < 1$ và $w_p+w_n+w_a = 1$.

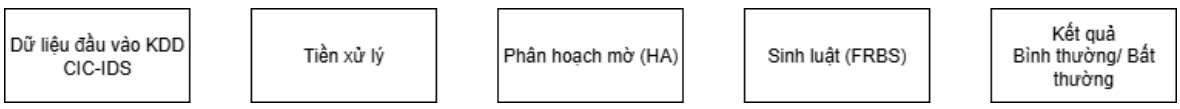
Trong phương pháp này, mỗi cá thể trong quần thể được sử dụng để mã hóa một hệ thống luật. Số lượng luật tối đa mà một cá thể có thể biểu diễn – hay số luật trong một hệ luật ứng viên – được giới hạn bởi một tham số định trước, ký hiệu là Nr_max. Do đó, mỗi cá thể sẽ bao gồm Nr_max biến đại diện cho mục tiêu chọn lọc luật, ký hiệu lần lượt là r_1, r_2, ..., r_{Nr_max}.

Mỗi biến r_i mang giá trị thực được chuẩn hóa trong khoảng [0,1]. Cách lựa chọn luật để đưa vào hệ luật cuối cùng được thực hiện dựa trên chỉ số vị trí, tính bằng công thức: $\lfloor r_i \times |R| \rfloor$, trong đó |R| là số lượng luật trong tập luật ứng cử, và $\lfloor . \rfloor$ biểu thị hàm làm tròn xuống (floor). Tập luật được chọn sẽ bao gồm các luật tại các chỉ số tương ứng này, với i chạy từ 1 đến Nr_max.

Quá trình tiến hóa của quần thể được thực hiện thông qua các phép toán di truyền cơ bản như lai ghép (crossover), đột biến (mutation), và chọn lọc (selection). Các phép toán này được triển khai tương tự như quy trình được trình bày trong tài liệu tham khảo [2].

3.2. XÂY DỰNG HỆ LUẬT MỜ PHÁT HIÊN BẤT THƯỜNG TRONG MẠNG MÁY TÍNH

Mô hình kiến trúc tổng thể của hệ thống



Hình 3.2. Mô hình kiến trúc tổng thể của hệ thống thực nghiệm

Mô hình hệ thống phát hiện bất thường trong mạng máy tính được thiết kế dựa trên nguyên lý của hệ suy diễn mờ (FRBS) kết hợp với cơ chế học tập và xử lý ngôn ngữ mờ thông qua đại số gia tử. Dữ liệu đầu vào được lấy từ các tập dữ liệu mạng tiêu chuẩn, qua bước tiền xử lý nhằm chuẩn hóa và trích

chọn các đặc trưng phù hợp. Sau đó, dữ liệu được đưa vào mô hình FRBS để thực hiện phân tích và phân loại.

Trong mô hình FRBS, các luật mờ được xây dựng dựa trên ngôn ngữ định lượng từ HA, giúp mô hình diễn đạt các khái niệm mờ như "cao", "thấp", "nguy cơ", "bình thường" một cách rõ ràng và có cơ sở toán học. Bộ phận học tập giúp tinh chỉnh hệ luật và hàm thành viên nhằm cải thiện hiệu quả suy diễn qua từng vòng huấn luyện.

Dữ liệu sau khi được xử lý qua FRBS sẽ được đánh giá mức độ bất thường và phân loại thành đầu ra tương ứng (bình thường hoặc bất thường). Mô hình này có khả năng thích nghi với dữ liệu phức tạp, không chắc chắn và mang lại hiệu suất ổn định trong các bài toán phát hiện tấn công mạng.

3.2.1 Chuẩn bị dữ liệu

Trong phần này tôi sử dụng hương pháp xây dựng hệ luật mờ trong mục 3.1 để tiến hành xây dựng các hệ mờ phát hiện bất thường trong mạng máy tính dựa trên các tập mẫu dữ liệu được lấy từ *http://cicresearch.ca/ CICDataset/ CIC-IDS-2017/Dataset/CIC-IDS-2017/CSVs/MachineLearningCSV.zip*. Tôi tiến hành xây dựng FRBS trên tập dữ liệu trên các tập dữ liệu mô tả ở bảng 3.1. Trong đó #Class là số lớp, #N là số mẫu dữ liệu bình thường, #AN số mẫu dữ liệu bất thường, #All tổng số mẫu dữ liệu.

Bảng 3.1. Danh sách các tập dữ liệu xây dựng FRBS

STT	**Tên tập dữ liệu**	**Loại hình tấn công**	**#Class**	**#N**	**#AN**	**#All**
1	*Friday-WorkingHours-Afternoon-DDos.pcap_ISCX*	DDos	2	97,690	128,025	225,715
2	Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX	PortScan	2	127,537	158,930	286,467
3	Friday-WorkingHours-Morning_Bot_.pcap_ISCX	Botnet	2	189,067	1,966	191,033

4	Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX	Infiltration	2	288,566	36	288,602

Các tập dữ liệu có 78 thuộc tính, chi tiết trong bảng dưới đây.

Bảng 3.2. Danh sách các thuộc tính của dữ liệu

STT	Tên thuộc tính	Ý nghĩa	STT	Tên thuộc tính	Ý nghĩa
1	Flow Duration	Thời gian lưu lượng	40	Fwd PSH Flags	Số lượng gói forward có cờ PSH được bật.
2	Destination Port	Cổng đến	41	Bwd PSH Flags	Số lượng gói backward có cờ PSH được bật.
3	Total Fwd Packets	Tổng số gói tin được gửi từ phía nguồn	42	Fwd URG Flags	Số lượng gói forward có cờ URG được bật.
4	Total Backward Packets	Tổng số gói tin được gửi từ phía đích	43	Bwd URG Flags	Số lượng gói backward có cờ URG được bật.
5	Total Length of Fwd Packets	Tổng độ dài của tất cả các gói tin forward.	44	Fwd Header Length	Tổng độ dài phần đầu gói (header) của forward packets.
6	Total Length of Bwd Packets	Tổng độ dài của tất cả các gói tin backward.	45	Bwd Header Length	Tổng độ dài phần đầu gói của backward packets.
7	Fwd Packet Length Max	Độ dài lớn nhất trong các gói tin forward.	46	Fwd Packets/s	Tốc độ forward packets mỗi giây.
8	Fwd Packet Length Min	Độ dài nhỏ nhất trong các gói tin forward.	47	Bwd Packets/s	Tốc độ backward packets mỗi giây.
9	Fwd Packet Length Mean	Độ dài trung bình của các gói tin forward.	48	Min Packet Length	Độ dài nhỏ nhất trong tất cả các gói tin.
10	Fwd Packet Length Std	Độ lệch chuẩn độ dài các gói tin forward.	49	Max Packet Length	Độ dài lớn nhất trong tất cả các gói tin.
11	Bwd Packet Length Max	Độ dài lớn nhất trong các gói tin backward.	50	Packet Length Mean	Độ dài trung bình của các gói tin.
12	Bwd Packet Length Min	Độ dài nhỏ nhất trong các gói tin backward.	51	Packet Length Std	Độ lệch chuẩn độ dài gói tin.

13	Bwd Packet Length Mean	Độ dài trung bình của các gói tin backward.	52	Packet Length Variance	Phương sai độ dài các gói tin.
14	Bwd Packet Length Std	Độ lệch chuẩn độ dài các gói tin backward.	53	FIN Flag Count	Số lượng gói có cờ FIN được bật.
15	Flow Bytes/s	Tốc độ truyền bytes mỗi giây trong luồng.	54	SYN Flag Count	Số lượng gói có cờ SYN được bật.
16	Flow Packets/s	Tốc độ truyền packets mỗi giây trong luồng.	55	RST Flag Count	Số lượng gói có cờ RST được bật.
17	Flow IAT Mean	Thời gian trung bình giữa các gói tin trong luồng.	56	PSH Flag Count	Số lượng gói có cờ PSH được bật.
18	Flow IAT Std	Độ lệch chuẩn thời gian giữa các gói trong luồng.	57	ACK Flag Count	Số lượng gói có cờ ACK được bật.
19	Flow IAT Max	Thời gian giữa hai gói tin lớn nhất trong luồng.	58	URG Flag Count	Số lượng gói có cờ URG được bật.
20	Flow IAT Min	Thời gian giữa hai gói tin nhỏ nhất trong luồng.	59	CWE Flag Count	Số lượng gói có cờ CWE được bật.
21	Fwd IAT Total	Tổng thời gian giữa các gói tin forward.	60	ECE Flag Count	Số lượng gói có cờ ECE được bật.
22	Fwd IAT Mean	Thời gian trung bình giữa các gói tin forward.	61	Down/Up Ratio	Tỷ lệ lưu lượng tải xuống so với tải lên.
23	Fwd IAT Std	Độ lệch chuẩn thời gian giữa các gói forward.	62	Average Packet Size	Kích thước trung bình của các gói tin.
24	Fwd IAT Max	Khoảng thời gian lớn nhất giữa các gói forward.	63	Avg Fwd Segment Size	Kích thước trung bình của các đoạn forward.
25	Fwd IAT Min	Khoảng thời gian nhỏ nhất giữa các gói forward.	64	Avg Bwd Segment Size	Kích thước trung bình của các đoạn backward.
26	Bwd IAT Total	Tổng thời gian giữa các gói tin backward.	65	Fwd Avg Bytes/Bulk	Trung bình số bytes mỗi khối dữ liệu forward.
27	Bwd IAT Mean	Thời gian trung bình giữa các gói tin backward.	66	Fwd Avg Packets/Bulk	Trung bình số gói tin mỗi khối dữ liệu forward.
28	Bwd IAT Std	Độ lệch chuẩn thời gian giữa các gói backward.	67	Fwd Avg Bulk Rate	Tốc độ trung bình của khối dữ liệu forward.

29	Bwd IAT Max	Khoảng thời gian lớn nhất giữa các gói backward.	68	Bwd Avg Bytes/Bulk	Trung bình số bytes mỗi khối dữ liệu backward.
30	Bwd IAT Min	Khoảng thời gian nhỏ nhất giữa các gói backward.	69	Bwd Avg Packets/Bulk	Trung bình số gói tin mỗi khối dữ liệu backward.
31	act_data_pkt_fwd	Số lượng gói tin dữ liệu thực sự được gửi đi từ phía nguồn.	70	Bwd Avg Bulk Rate	Tốc độ trung bình của khối dữ liệu backward.
32	min_seg_size_forward	Kích thước đoạn nhỏ nhất trong forward.	71	Subflow Fwd Packets	Số lượng gói tin forward trong subflow.
33	Active Mean	Thời gian trung bình giữa các hoạt động truyền tải.	72	Subflow Fwd Bytes	Tổng số byte forward trong subflow.
34	Active Std	Độ lệch chuẩn thời gian hoạt động truyền tải.	73	Subflow Bwd Packets	Số lượng gói tin backward trong subflow.
35	Active Max	Khoảng thời gian hoạt động dài nhất.	74	Subflow Bwd Bytes	Tổng số byte backward trong subflow.
36	Active Min	Khoảng thời gian hoạt động ngắn nhất.	75	Init_Win_bytes_forward	Kích thước cửa sổ ban đầu forward.
37	Idle Mean	Thời gian nhàn rỗi trung bình giữa các luồng.	76	Init_Win_bytes_backward	Kích thước cửa sổ ban đầu backward.
38	Idle Std	Độ lệch chuẩn thời gian nhàn rỗi.	77	Idle Min	Thời gian nhàn rỗi ngắn nhất giữa các luồng.
39	Idle Max	Thời gian nhàn rỗi dài nhất giữa các luồng.	78	Label	Nhãn của mẫu dữ liệu

Tập dữ liệu sẽ được chia ngẫu nhiên thành 2 phần 90% để huấn luyện (data training) dùng để xây dựng FRBS, 10% để kiểm tra (data test). Mô tả các tập dữ liệu huấn luyện, kiểm tra tra trong bảng dưới đây.

Bảng 3.3. Số mẫu dữ liệu huấn luyện và kiểm tra

STT	**Tập dữ liệu**	**Tập huấn luyện (10%)**		**Tập kiểm tra (90%)**		
		#N	**#AN**	**#N**	**#AN**	**#All**
1	DDos	114,999	88,145	13,026	9,545	225,715
2	PortScan	11,733	15,893	115,804	143,037	286,467

3	Botnet	17,961	197	171,106	1,769	191,033
4	Infiltration	27,413	4	261,153	32	288,602

3.2.2. Thiết lập các tham số thí nghiệm

Quá trình xây dựng FRBS được thực hiện gồm hai pha:

- **Pha thứ nhất:** Tìm bộ tham số tính mờ tối ưu của HA bằng thuật toán OP-PARHA, ở pha này để tính giá trị hàm mục tiêu chúng ta thực hiện với phương pháp thử nghiệm *Test-All*. Thiết lập các tham số thực hiện pha thứ nhất trong bảng 3.4 dưới đây.

Bảng 3.4. Thiết lập các tham số tối ưu hóa OP-PARHA

	Tham số	**Giá trị**
Nhóm tham số thuật giải di truyền	- Xác xuất lại ghép - Xác suất đột biến - Chiều dài mỗi gen - Số cá thể của quần thể - Số thế hệ tiến hóa	*Pcross* = 0.85 *Pmu* = 0.05 *lchrom* = 24 *Populations* = 300 *NGenerations* = 200
Tham số của đại số gia tử	Tham số độ đo tính mờ: - Phần tử sinh c- - Của gia tử *L* - Độ dài tối đa của từ k_j	 $0.3 \le \mu^j_{fmc-} \le 0.7$ với $j = 1,..,n$ $0.3 \le \mu^j_L \le 0.7$ với $j = 1,..,n$ $1 \le k_j \le 3$ với $j = 1,..,n$
Tham số thuật toán SGERD	- Tham số ngưỡng ρ - Chiều dài tối đa của luật	$0.3 \le \rho \le 0.7$ với $j = 1,..,n$ $l_{max} = 15$

Trong nghiên cứu [19], Mansoori và cộng sự đã chỉ ra rằng giá trị của tham số *Q* có vai trò quyết định đến độ chính xác của các hệ luật được xây dựng. Các tác giả đã lựa chọn giá trị *Q* dựa trên ý tưởng rằng kích thước của quần thể chính và quần thể bổ trợ phải bằng một nửa số luật được khởi tạo ban đầu có độ dài 1.

Do đó, Q được tính toán theo công thức: $Q = \frac{\sum_{j=1}^{n}|X_{j(k_j)}|}{2\times m}$. Trong đó: $|X_{j(kj)}|$ là số lượng từ ngôn ngữ được sử dụng cho biến X_j, n là số thuộc tính (chiều dữ liệu), m là số lớp (trong nghiên cứu này m=2).

Tuy nhiên, với các tập dữ liệu có số chiều lớn, giá trị của Q có thể trở nên rất lớn, dẫn đến tăng đáng kể thời gian tính toán của thuật toán. Để khắc phục nhược điểm này, giá trị của Q được hạn chế không vượt quá 20. Điều này giúp kiểm soát chi phí tính toán mà vẫn duy trì được hiệu quả tìm kiếm, như vậy Q được xác định theo công thức sau đây $Q = \min(\frac{\sum_{j=1}^{n}|X_{j(k_j)}|}{2\times m}, 20)$.

- Pha thứ hai: Hệ luật tối ưu được xác định bằng thuật toán HA-OFRB, áp dụng trên tập luật ứng cử R do thuật toán HA-SGERD sinh ra. Tập luật này được tạo ra dựa trên bộ tham số tính mờ của HA và ngưỡng, cả hai đều đã được tối ưu trong pha thứ nhất của quy trình.

3.2.3. Kết quả thí nghiệm

Để đánh giá hiệu quả của mô hình phát hiện bất thường, đề tài sử dụng một số độ đo phổ biến trong lĩnh vực học máy và an ninh mạng, bao gồm:

- **Accuracy (Độ chính xác)**: Phản ánh tỷ lệ tổng thể các dự đoán đúng trên toàn bộ tập dữ liệu.
- **Precision**: Cho biết trong số các mẫu được mô hình dự đoán là bất thường, có bao nhiêu mẫu thực sự là bất thường.
- **Recall (Tỷ lệ phát hiện)**: Đo lường khả năng của mô hình trong việc phát hiện đúng tất cả các trường hợp bất thường thực sự có trong dữ liệu.
- **F1-Score**: Là trung bình điều hòa giữa Precision và Recall, giúp cân bằng giữa hai tiêu chí này trong trường hợp dữ liệu mất cân đối.

Các chỉ số này được tính dựa trên ma trận nhầm lẫn (confusion matrix) và được sử dụng để đánh giá toàn diện khả năng phân loại của mô hình trong cả hai chiều: phát hiện chính xác và tránh báo động sai.

Kết quả thuật toán HA-OFRB được tổng hợp trong các bảng 3.5, 3.6.

Bảng 3.5. Độ chính Acc, TP, FP, TN và FN trên tập huấn luyên và trên tập kiểm tra

STT	Tập dữ liệu	Tập kiểm tra (10%)					Tập huấn luyện (90%)				
		TP	FP	TN	FN	ACC	TP	FP	TN	FN	ACC
1	DDos	11,250	1,013	8,756	1,552	88.64%	97,453	6,597	81,324	17,770	88.01%
2	PortScan	14,321	702	11,031	1,572	91.77%	137,867	12,479	103,325	5,170	93.18%
3	Botnet	182	1,727	16,234	15	90.41%	1,653	7,684	163,422	116	95.49%
4	Infiltration	3	3,099	24,314	1	88.69%	28	7,888	253,265	4	96.98%

Bảng 3.6. Độ đo Precision, Recall, F1-Score của các hệ luật

STT	Tập dữ liệu	Tập kiểm tra (10%)			Tập huấn luyện (90%)		
		Precision	Recall	F1-score	Precision	Recall	F1-score
1	DDos	0.92	0.88	0.90	0.94	0.85	0.89
2	PortScan	0.95	0.90	0.93	0.92	0.96	0.94
3	Botnet	0.10	0.92	0.17	0.18	0.93	0.30
4	Infiltration	0.001	0.75	0.002	0.004	0.88	0.007

Kết quả thử nghiệm mô hình Fuzzy Rule-Based System (FRBS) trên các loại tấn công mạng khác nhau trình bày trong bảng 3.6 cho thấy khả năng phát hiện tấn công của mô hình là không đồng đều, phản ánh sự phụ thuộc vào đặc trưng của từng loại tấn công. Cụ thể, đối với hai loại tấn công phổ biến là DDoS và PortScan, mô hình FRBS hoạt động tương đối hiệu quả. Các chỉ số như Precision, Recall và F1-score đều đạt mức cao (trên 0.89), chứng minh khả năng nhận diện chính xác và ổn định. Điều này cho thấy FRBS có thể học tốt

các mẫu đặc trưng rõ rệt và lặp lại trong dữ liệu, đặc biệt là các cuộc tấn công có tần suất lớn và dấu hiệu nhận biết rõ ràng như DDoS và PortScan.

Tuy nhiên, đối với các dạng tấn công tinh vi hơn như Botnet và Infiltration, hiệu quả của mô hình giảm đáng kể. Trong trường hợp Botnet, mặc dù Recall đạt mức rất cao (trên 0.92) – cho thấy hầu hết các cuộc tấn công đều được phát hiện – nhưng Precision lại cực kỳ thấp (dưới 0.1), dẫn đến F1-score thấp (khoảng 0.17). Điều này phản ánh tình trạng mô hình sinh ra quá nhiều cảnh báo sai, thiếu khả năng phân biệt chính xác. Với Infiltration, tình trạng còn nghiêm trọng hơn khi Precision gần bằng 0 và F1-score cũng gần 0, cho thấy mô hình gần như không thể nhận diện loại tấn công này.

Kết quả này cho thấy FRBS, với đặc điểm dựa vào luật mờ định nghĩa thủ công hoặc học từ dữ liệu, có thể phù hợp với những trường hợp tấn công rõ ràng nhưng gặp khó khăn khi xử lý các mẫu không phổ biến hoặc phức tạp. Cần nghiên cứu các hướng cải tiến như tối ưu hóa bộ luật, bổ sung đặc trưng, sử dụng hybrid với các mô hình học sâu, hoặc điều chỉnh cơ chế suy diễn để cải thiện hiệu suất tổng thể của FRBS trong môi trường thực tế.

3.3. KẾT LUẬN CHƯƠNG 3

Trong chương này luận án đã trình bày phương pháp xây dựng FRBS dựa trên đại số gia tử để giải bài toán phân lớp nói chung và áp dụng vào xây dựng FRBS phát hiện bất thường trong mạng máy tính. Thuật toán đề xuất trong [4] được trình bày lại và được thử nghiệm trên 4 tập dữ liệu trong tập dữ liệu CIC-IDS-2017. Kết quả thử nghiệm cho thấy độ chính xác của FRBS đạt thấp nhất là 88% trên cả huấn luyện và tập kiểm tra. Dựa trên các chỉ số Precision, Recall, F1-Score cho thấy phương pháp sử dụng FRBS trong phát hiện bất thường trong mạng máy tính hoạt động tốt với tập mẫu dữ liệu có đặc trưng rõ rệt và lặp lại trong dữ liệu, đặc biệt là các cuộc tấn công có tần suất lớn và dấu hiệu

nhận biết rõ ràng như DDoS và PortScan, đối với các dạng tấn công tinh vi hơn như Botnet và Infiltration, hiệu quả của mô hình giảm đáng kể.

KẾT LUẬN

Trong bối cảnh an ninh mạng ngày càng phức tạp với sự gia tăng nhanh chóng của các mối đe dọa tiên tiến, đặc biệt khi trí tuệ nhân tạo vừa thúc đẩy khả năng tấn công tinh vi, vừa nâng cao năng lực phòng thủ, việc phát hiện bất thường trong mạng máy tính trở nên cấp thiết hơn bao giờ hết. Đề án "Nghiên cứu ứng dụng mô hình mờ và lý thuyết đại số gia tử phát hiện bất thường trong mạng máy tính" đã tập trung giải quyết bài toán này, đặc biệt nhấn mạnh vào khả năng phát hiện các cuộc tấn công mới, chưa từng xuất hiện, điều mà các hệ thống dựa trên dấu hiệu truyền thống còn hạn chế.

Mục tiêu chính của đề án là nghiên cứu sâu về các phương pháp phát hiện bất thường trong mạng máy tính, với trọng tâm là phát triển một hệ thống dựa trên mô hình mờ (FRBS) và lý thuyết đại số gia tử. Lý do lựa chọn hướng tiếp cận này xuất phát từ ưu điểm nổi bật của FRBS như chi phí tính toán thấp, dễ hiểu và khả năng mô phỏng gần gũi phương pháp suy luận của con người, rất phù hợp cho việc phân loại trạng thái mạng theo thời gian thực.

Trong quá trình nghiên cứu, đề án đã thực hiện khảo sát các khái niệm cơ bản về phát hiện bất thường trong mạng máy tính, bao gồm định nghĩa, phân loại các dạng tấn công (U2R, R2L, DoS, Probe) và khung kiến trúc tổng thể của mô hình NAD. Các phương pháp phát hiện bất thường phổ biến đã được phân tích chi tiết, từ các phương pháp truyền thống dựa trên thống kê, khai phá dữ liệu, học có giám sát (ANN, SVM, KNN, DT) đến các phương pháp học không giám sát (K-means) và bán giám sát (OCSVM, LOF, KDE). Đặc biệt, đề án cũng đã đánh giá các phương pháp phân lớp đơn dựa trên học sâu như AutoEncoder (AE) và Shrink AutoEncoder (SAE), nhận thấy tiềm năng vượt trội của chúng đối với dữ liệu lớn và phức tạp, đồng thời chỉ ra những hạn chế cố hữu cần được khắc phục. Các độ đo đánh giá hiệu quả của mô hình như: Độ chính xác, Ma trận lỗi, Detection Rate (DR), False Alarm Rate (FAR), F1-

Score, Đường cong ROC và AUC cũng được trình bày rõ ràng, tạo cơ sở cho việc đánh giá khách quan các mô hình được đề xuất.

Chương hai của đề án đã đi sâu vào nền tảng lý thuyết của mô hình mờ và đại số gia tử, trình bày các kiến thức cơ bản về tập mờ và cách xây dựng các hệ phân lớp mờ. Đây là cơ sở lý thuyết quan trọng để xây dựng FRBS hiệu quả. Cuối cùng, Chương ba trình bày phương pháp xây dựng hệ luật mờ dựa trên đại số gia tử, cùng với việc mô tả tập dữ liệu được sử dụng để phát hiện tấn công DDoS, PortScan, Botnet, Infiltration phân tích và đánh giá kết quả huấn luyện mô hình.

Đề án đã xây dựng một nền tảng vững chắc cho việc nghiên cứu và ứng dụng mô hình mờ kết hợp lý thuyết đại số gia tử trong phát hiện bất thường mạng. Các kết quả nghiên cứu lý thuyết và phương pháp luận đã cho thấy tiềm năng của hướng tiếp cận này trong việc cung cấp một giải pháp NAD hiệu quả, có khả năng xử lý các loại tấn công mới và nâng cao năng lực phòng thủ an ninh mạng trong kỷ nguyên số.

HƯỚNG PHÁT TRIỂN

Dựa trên những kết quả và phân tích trong đề án, các hướng nghiên cứu tiếp theo có thể bao gồm:

- *Tối ưu hóa và cải tiến mô hình:* Tiếp tục nghiên cứu và phát triển các thuật toán xây dựng FRBS dựa trên đại số gia tử để tối ưu hóa hiệu suất, giảm chi phí tính toán và nâng cao khả năng phát hiện các loại bất thường phức tạp, đặc biệt là những loại mà các mô hình hiện tại như SAE còn gặp khó khăn (ví dụ R2L).
- *Kết hợp với các công nghệ tiên tiến khác*: Nghiên cứu khả năng tích hợp FRBS với các công nghệ học sâu khác hoặc các kỹ thuật tiền xử lý dữ liệu tiên tiến để tận dụng ưu điểm của từng phương pháp, tạo ra một hệ thống NAD lai ghép mạnh mẽ hơn.

- *Triển khai thực nghiệm:* Xây dựng một hệ thống NAD thử nghiệm hoàn chỉnh để kiểm chứng và đánh giá hiệu quả của mô hình trong môi trường mạng thực tế, góp phần đưa kết quả nghiên cứu vào ứng dụng thực tiễn.
- *Phát triển giao diện người dùng:* Xây dựng giao diện trực quan để người quản trị mạng dễ dàng theo dõi, cấu hình và nhận cảnh báo từ hệ thống, nâng cao tính ứng dụng và khả năng vận hành.

TÀI LIỆU THAM KHẢO

Tiếng Việt

[1] Nguyễn Hà Dương và Hoàng Đăng Hải (2016), "Phát hiện lưu lượng mạng bất thường trong điều kiện dữ liệu huấn luyện chứa ngoại lai", *Tạp chí Khoa học Công nghệ Thông tin và Truyền thông - Học viện Công nghệ Bưu chính Viễn thông*, tr. 03–16.

[2] Hoàng Kiếm, Lê Hoàng Thái, Giải thuật di truyền - Cách giải tự nhiên các bài toán trên máy tính, Nhà Xuất bản giáo dục, năm 2000.

[3] Dương Thăng Long, Nguyễn Cát Hồ, Trần Thái Sơn, "Một phương pháp xây dựng hệ luật mờ có trọng số để phân lớp dựa trên đại số gia tử", Tạp chí Tin học và Điều khiển học, T.26(1)(2010) trang 55-72.

[4] Nguyễn Cát Hồ, Hoàng Văn Thông, Nguyễn Văn Long, Một phương pháp tiến hóa sinh hệ luật mờ với ngữ nghĩa thứ tự ngôn ngữ, Tin học và điều khiển học, Tập 28 số 4, 2012, trang 333-345.

Tiếng Anh

[5] Bernhard Scho¨lkopf et al. (2001), "Estimating the support of a high dimensional distribution", Neural computation, 13 (7), pp. 1443–1471.

[6] O. Cordón, M. J. del Jesus, F. Herrera, A proposal on reasoning methods in fuzzy rule-based classification systems. Int. J. Approx. Reason. 20(1) (1999) pp. 21–45.

[7] David L Olson and Dursun Delen (2008), Advanced data mining techniques, Springer Science & Business Media.

[8] Douglas M Hawkins (1980), *Identification of outliers*, vol. 11, Springer.

[9] M. J. Gacto, R. Alcalá, F. Herrera, Adaptation and Application of Multi-Objective Evolutionary Algorithms for Rule Reduction and Parameter Tuning of Fuzzy Rule-Based Systems, Soft Comput., Volume 13, Issue 5 (2008) pp. 419-443.

[10] M.J. Gacto, R. Alcalá, F. Herrera, Interpretability of Linguistic Fuzzy Rule-Based Systems: An Overview of Interpretability Measures. Inform. Sci., 181:20 (2011) pp. 4340–4360.

[11] Geoffrey E Hinton and Richard S Zemel, “Autoencoders, minimum description length and Helmholtz free energy”, in: Advances in neural information processing systems, 1994, pp. 3–10.

[12] Guoquan Li et al. (2018), “Data Fusion for Network Intrusion Detection: A Review”, Security and Communication Networks, 2018, pp. 1– 16, doi: 10.1155/2018/8210614.

[13] Igr Alexánder Fernández-Saúco et al., “Computing Anomaly Score Threshold with Autoencoders Pipeline”, in: Iberoamerican Congress on Pattern Recognition, Springer, 2018, pp. 237–244.

[14] H. Ishibuchi, K. Nozaki, N. Yamamoto, H. Tanaka, Selecting fuzzy if-then rules for classification problems using genetic algorithms. IEEE Trans. Fuzzy Syst. 3(3) (1995) pp. 260–270.

[15] H. Ishibuchi, Y. Nojima, Analysis of interpretability-accuracy tradeoff of fuzzy systems by multiobjective fuzzy genetics-based machine learning, Int. J. Approx. Reason., vol.44, no.1 (2007) pp. 4–31.

[16] J. D. Knowles and D.W. Corne, Approximating the non dominated front using the Pareto archived evolution strategy, Evol. Comput., vol. 8, no. 2 (2000) pp. 149–172.

[17] Lukas Ruff et al., “Deep one-class classification”, in: International Conference on machine learning, 2018, pp. 4393–4402.

[18] Marina Sokolova, Nathalie Japkowicz, and Stan Szpakowicz, “Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation”, in: Australasian joint conference on artificial intelligence, Springer, 2006, pp. 1015–1021.

[19] E.G. Mansoori, M.J. Zolghadri, and S.D. Katebi, SGERD: A Steady-Sate Genetic Algorithm for Extracting Fuzzy Classification Rules From Data, IEEE Trans. on fuzzy syst., Vol 16, No. 4 (2008), pp. 1061-1071.

[20] Markus M Breunig et al., “LOF: identifying density-based local outliers”, in: *ACM sigmod record*, vol. 29, 2, ACM, 2000, pp. 93–104.

[21] Martin Roesch et al., “Snort: Lightweight intrusion detection for networks.”, in: *Lisa*, vol. 99, 1, 1999, pp. 229–238.

[22] Matt P Wand and M Chris Jones (1994), *Kernel smoothing*, Chapman and Hall/CRC.

[23] C. Mencar, A.M. Fanelli, Interpretability constraints for fuzzy information granulation, Inform. Sci. 178 (2008) pp. 4585–4618.

[24] C. Mencar, C. Castiello, R. Cannone, A.M. Fanelli, Interpretability assessment of fuzzy knowledge bases: a cointension based approach, Int. J. Approx. Reason. 52 (2011) pp. 501–518.

[25] G.A. Miller, The magical number seven plus or minus two: some limits on our apacity for processing information, The Psychological Review 63 (1956), pp. 81–97.

[26] Mohiuddin Ahmed and Abdun Naser Mahmood, "Network traffic analysis based on collective anomaly detection", in: *2014 9th IEEE Conference on Industrial Electronics and Applications*, IEEE, 2014, pp. 1141– 1146.

[27] Mohiuddin Ahmed, Abdun Naser Mahmood, and Jiankun Hu (2016), "A survey of network anomaly detection techniques", *Journal of Network and Computer Applications*, 60, pp. 19–31.

[28] D. Nauck, Measuring interpretability in rule-based classification systems, in: Proceed. of the 12th IEEE Int. Conf. on Fuzzy Syst., vol. 1 (2003) pp. 196–201

[29] C.H. Nguyen and W. Wechler, Hedge algebras: an algebraic approach to structures of sets of linguistic domains of linguistic truth variables, Fuzzy Sets and Syst., 35(3) (1990) pp. 281-293.

[30] C. H. Nguyen and W. Wechler, Extended algebra and their application to fuzzy logic, Fuzzy Sets and Syst., vol.52 (1992) pp. 259–281.

[31] C. H. Nguyen, A topological completion of refined hedge algebras and a model of fuzziness of linguistic terms and hedges, Fuzzy Sets and Syst., vol.158 (2007) pp.436-451.

[32] C. H. Nguyen and V. L. Nguyen, Fuzziness measure on complete hedges algebras and quantifying semantics of terms in linear hedge algebras, Fuzzy Sets and Syst., vol.158 (2007) pp.452-471.

[33] C. H. Nguyen, W. Pedryczb, T. L. Duong, T. S. Tran, A genetic deSg of linguistic terms for fuzzy rule based classifiers, Int. J. Approx. Reason., 54 (2013) 1–2.1

[34] Prasanta Gogoi et al. (2011), “A survey of outlier detection methods in network anomaly identification”, The Computer Journal, 54 (4), pp.570–588.

[35] P. Pulkkinen and H. Koivisto, Fuzzy classifier identification using decision tree and multiobjective evolutionary algorithms, Int. J. Approx. Reason., vol. 48, no. 2 (2008) pp. 526–543.

[36] Raghavendra Chalapathy and Sanjay Chawla (2019), “Deep Learning for Anomaly Detection: A Survey”, arXiv, arXiv–1901.

[37] Sarah M Erfani et al. (2016), “High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning”, Pattern Recognition, 58, pp. 121–134.

[38] Van Loi Cao (2018), “Improving Network Anomaly Detection with Genetic Programming and Autoencoders”.

[39] L. A. Zadeh, Fuzzy set, Information and control, 8, (1965), pp. 338-353

[40] L. A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning, Parts I, II and III. Inform. Sci. 8, 8, 9 (1975), pp 199–249, pp. 301–357, pp. 43–80.