HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

BÙI CÔNG THÀNH

PHÁT TRIỂN MỘT SỐ MÔ HÌNH PHÁT HIỆN BẤT THƯỜNG MẠNG DỰA TRÊN HỌC SÂU VÀ TỔNG HỢP DỮ LIỆU

LUẬN ÁN TIẾN SĨ KỸ THUẬT

HÀ NỘI – 2021

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

BÙI CÔNG THÀNH

PHÁT TRIỂN MỘT SỐ MÔ HÌNH PHÁT HIỆN BẤT THƯỜNG MẠNG DỰA TRÊN HỌC SÂU VÀ TỔNG HỢP DỮ LIỆU

CHUYÊN NGÀNH: HỆ THỐNG THÔNG TIN
MÃ SỐ: : 9.48.01.04

LUẬN ÁN TIẾN SĨ

NGƯỜI HƯỚNG DẪN KHOA HỌC:
1. PGS.TS. HOÀNG MINH
2. PGS. TS. NGUYỄN QUANG UY

HÀ NỘI – 2021

TÓM TẮT

Sự phát triển nhanh của mạng máy tính và IoT (sau đây gọi là mạng) cả về dịch vụ và hạ tầng đã kéo theo những thách thức rất lớn trong vấn đề bảo đảm an ninh mạng. Tìm kiếm giải pháp phát hiện các tấn công mạng là nhiệm vụ trọng tâm cho bảo vệ an ninh mạng, trong đó phát hiện bất thường mạng (Network Anomaly Detection -NAD) được rất nhiều các học giả quan tâm nghiên cứu trong những năm qua. NAD là lĩnh vực nghiên cứu để tìm ra các giải pháp hiệu quả trong phân tách giữa trạng thái bình thường và bất thường mạng. Học máy được biết như phương pháp chủ yếu cho xây dựng các thuật toán phát hiện bất thường. Các mô hình học máy được huấn luyện chỉ với dữ liệu bình thường hay còn gọi là các bộ phân đơn lớp (One-class Classification - OCC) được cho là sự lựa chọn phù hợp và đang cho thấy các kết quả phát hiện bất thường rất hiệu quả. Những năm gần đây, phát triển các kỹ thuật học sâu (deep learning) đã mạng lại nhiều thành tựu trong các lĩnh vực, học sâu dựa trên kiến trúc AutoEncoders (AE) được công nhận rộng rãi là phương pháp tiên tiến, có khả năng giải quyết các vấn đề phức tạp của phát hiện bất thường mạng, tiêu biểu trong đó là SAE (Shrink AutoEncoder).

Mặc dù vậy, các phương pháp NAD cần phải liên tục được nghiên cứu cải tiến để có thể đáp ứng tốt hơn khi mà các nguy cơ đe doạ an ninh mạng ngày càng tăng. Thêm vào đó, các phương pháp NAD đơn lẻ dựa trên OCC nhìn chung đang phải đối mặt với một số thách thức khác như: mỗi phương pháp đơn được cho là chỉ hiệu quả trên một điều kiện môi trường mạng cụ thể; các phương pháp OCC vẫn cần sự hỗ trợ của chuyên gia để đưa ra ngưỡng quyết định, đây là yêu cầu đối với một mô hình phát hiện tấn công khi được triển khai trong thực tế.

Luận án hướng tới mục tiêu nghiên cứu cải tiến phương pháp phát hiện bất thường mạng theo hướng giải quyết một số vấn đề đặt ra trên. Kết quả một số

nội dung chính đã được thực hiện gồm. (i) Đã đề xuất được giải pháp cho cải tiến một số hạn chế của phương pháp học sâu NAD tiêu biểu, các thuật toán cải tiến cho phép xây dựng mô hình NAD hiệu quả hơn trong điều kiện dữ liệu của đối tượng quan sát có tính phân cụm cao, tồn tại ở dạng nhiều cụm; có thể phát hiện hiệu quả hơn đối với nhóm tấn công mạng mà mô hình tiêu biểu dựa trên học sâu AutoEncoder gặp khó. (ii) Luận án đã đề xuất được mô hình khung tổng hợp dữ liệu, có tên OFuseAD, cho bài toán phát hiện bất thường. Mô hình đạt được từ kết quả cải tiến lý thuyết Dempster-Shafer, giải quyết các thách thức trong kết hợp các phương pháp OCC như xác định ngưỡng, trọng số cho kết hợp, cơ sở chọn lựa phương pháp đơn tham gia mô hình tổng hợp.

Kết quả thử nghiệm mô hình OFuseAD trên mười tập dữ liệu phổ biến trong lĩnh vực an ninh mạng cho thấy mô hình hoạt động khả thi, cho hiệu quả phát hiện bất thường hiệu quả, ổn định hơn so với các phương pháp đơn OCC trong đa số tập dữ liệu (9/10 tập dữ liệu thực nghiệm). Ngoài ra, mô hình OFuseAD có thể hoạt động mà không cần sự can thiệp cuả chuyên gia trong thiết lập ngưỡng quyết định.

Các vấn đề trên đã được luận án nghiên cứu, giải quyết. Các đóng góp của luận án đã được công bố trong các công trình khoa học có uy tín. Trong hiểu biết của nghiên cứu sinh, đóng góp của luận án mới và không trùng với các kết quả nghiên cứu đã công bố trong và ngoài nước.

LỜI CAM ĐOAN

Tôi xin cam đoan rằng nội dung luận án là kết quả nghiên cứu đã được thực hiện bởi tác giả dưới sự hướng dẫn của các thầy hướng dẫn khoa học. Luận án sử dụng các trích dẫn thông tin từ nhiều nguồn khác nhau và có nguồn gốc rõ ràng. Những đóng góp trong luận án đã được công bố trong các bài báo của tác giả và chưa được công bố trên bất kỳ công trình khoa học nào khác.

Hà Nội, ngày...tháng...năm 2021

LỜI CẢM ƠN

Thực hiện luận án Tiến sĩ đòi hỏi nghiên cứu sinh phải tập trung cao độ, trong thời gian dài. Kết quả nghiên cứu của NCS là sự góp sức rất lớn từ các thầy hướng dẫn khoa học, cơ sở đào tạo, cơ quan công tác, đồng nghiệp và đặc biệt là gia đình. Tôi muốn bày tỏ lòng biết ơn đối với họ.

Nghiên cứu sinh xin được bày tỏ lòng biết ơn sâu sắc đến Thầy giáo PGS.TS. Hoàng Minh và PGS.TS. Nguyễn Quang Uy đã tận tình hướng dẫn, trang bị kiến thức khoa học và phương pháp nghiên cứu để tôi hoàn thành nội dung nghiên cứu luận án. Tôi xin cảm ơn TS. Cao Văn Lợi về những góp ý rất hữu ích, giúp tôi thêm động lực trong nghiên cứu.

Nghiên cứu sinh xin bày tỏ lòng biết ơn chân thành tới Học viện Công nghệ Bưu chính Viễn thông, Khoa Sau đại học, các thầy cô giáo đã giúp đỡ tôi trong suốt quá trình tham gia học tập. Nghiên cứu sinh xin bày tỏ lòng biết ơn đến BTL Thông tin liên lạc, các Thủ trưởng và đồng chí tại Trung tâm Kỹ thuật thông tin công nghệ cao đã giúp đỡ, tạo điều kiện thời gian cho tôi.

Cuối cùng, nghiên cứu sinh vô cùng biết ơn đến gia đình bạn bè và người thân, bố mẹ hai bên đã luôn động viên khích lệ tôi, vợ tôi Đặng Thị Bích đã luôn cổ vũ động viên, chăm sóc gia đình và các con để tôi yên tâm nghiên cứu hoàn thành luận án.

NCS. Bùi Công Thành

MỤC LỤC

DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT

Viết tắt	**Viết đầy đủ**	**Nghĩa**
ACC	Accuracy	Chỉ số độ chính xác
AD	Anomaly Detection	Phát hiện bất thường
AE	AutoEncoder	Kiến trúc mạng nơ-ron AutoEncoder
ANN	Artificial Neural Network	Mạng nơ-ron nhân tạo
AS	Anomaly Score	Độ đo bất thường
BPA	Basic Probability Assignment	Hàm gán trọng số cơ bản của lý thuyết D-S
AUC	Area Under the Curve	Chỉ số đo dựa trên diện tích dưới đường cong ROC
Bayes	A Bayesian Inference	Suy luận Bayes
CEN	Centroid	Thuật toán Centroid
CNN	Convolution Neural Network	Mạng nơ-ron tích chập
KSAE	Clustering-Shrink Autoencoder	Mô hình kết hợp phân cụm và SAE
CTU	Czech Technical University	Đại học kỹ thuật Séc
DAE	Denoising Autoencoder	Mạng giảm nhiễu AE
DARPA	Defence Advanced Research Project Agency	Tổ chức DARPA
DBN	Deep Belief Network	Mạng niềm tin theo học sâu
DeAE	Deep AutoEncoder	Mạng nơ-ron học sâu AE

Viết tắt	**Viết đầy đủ**	**Nghĩa**
DF	Data Fusion	Tổng hợp dữ liệu
DoS	Denial of Service	Từ chối dịch vụ
DSAE	Double-Shrink AutoEncoder	Mô hình phát hiện bất thường DSAE
DTh	Decision Threshold	Ngưỡng quyết định
D-S	Dempster Shafer	Lý thuyết ra quyết định dựa trên dẫn chứng
DRC	Dempster Shafer Rule Combination	Hàm kết hợp của lý thuyết D-S
DR	Detection Rate	Chỉ số độ đo tỉ lệ phát hiện đúng
F1	F1-score	Chỉ số độ đo F1
FAR	False Alarm Rate	Chỉ số độ đo tỉ lệ phát hiện sai
F-SVDD	Fast Support Vector Data Description	Mô tả dữ liệu vector hỗ trợ tốc độ cao
FoD	Frame of Discernment	Tập giả thuyết trong lý thuyết D-S
FN	False Negative	Âm tính giả
FP	False Positive	Dương tính giả
FtR	Feature Representation	Đại diện đặc trưng
FuseNAD	Fusion-based Network Anomaly Detection towards Evidence Theory	Phương pháp phát hiện bất thường dựa trên tổng hợp dữ liệu sử dụng lý thuyết D-S
GA	Genetic Algorithm	Thuật toán di truyền
GMM	Gaussian Mixture Model	Mô hình hỗn hợp Gauss
GP	Genetic Programming	Lập trình di truyền
GS	Generalization Score	Độ đo tính khái quát hoá

Viết tắt	**Viết đầy đủ**	**Nghĩa**
HIDS	Host base IDS	IDS cài đặt trên các máy tính
HighDOD	High-dimensional Outlying Subspace Detection	Phát hiện điểm cá biệt trong không gian con nhiều chiều
IDS	Intrusion Detection System	Hệ thống phát hiện xâm nhập
KDD	Knowledge Discovery and Data Mining Tools Competition	Giải thi thường niên về khám phá tri thức và khai phá dữ liệu
KDE	Kernel Density Estimation	Phương pháp ước lượng dựa trên mật độ
K-NN	K-Nearest Neighbors	K láng giềng gần nhất
LOF	Local Outlier Factor	Phương pháp phát hiện bất thường dựa vào yếu tố cục bộ
MSE	Mean Square Error	Sai số toàn phương trung bình
NAD	Network Anomaly Detection	Phát hiện bất thường mạng
NIDS	Network Intrusion Detection System	Hệ thống phát hiện xâm nhập mạng
NLP	Natural Language Processing	Xử lý ngôn ngữ tự nhiên
OCC	One-class Classification	Phân đơn lớp
OCCNN	One Class Neural Network	Mạng nơ-ron hướng OCC
OCSVM	One-class Support Vector Machine	Máy véc tơ hỗ trợ một lớp

Viết tắt	**Viết đầy đủ**	**Nghĩa**
OFusedAD	One-class Fusion-based Anomaly Detection Framework	Mô hình khung phát hiện bất thường dựa trên tổng hợp dữ liệu từ các phương pháp OCC, sử dụng lý thuyết D-S
One-hot	One-hot Encoder	Mã hoá nhị phân (bit) hoá dữ liệu
PCA	Principal Component Analysis	Phép phân tích thành phần chính
R2L	Remote to Local	Tấn công từ xa vào nội bộ
RE	Reconstruction Error	Sai số tái tạo
ROC	Receiver Operating Characteristic	Chỉ số cho đánh giá mô hình phân lớp sử dụng đường cong ROC
SAE	Shrink AutoEncoder	Phương pháp co SAE
SGD	Stochastic Gradient Descent	Đạo hàm lặp giảm dần
SglAD	Single Anomaly Detection	Phương pháp đơn phát hiện bất thường
SOM	Self-Organizing Maps	Bản đồ tự tổ chức
SVDD	Support Vector Data Description	Mô tả dữ liệu vector hỗ trợ
SVM	Support Vector Machine	Máy vector hỗ trợ
U2R	User to Root	Loại tấn công leo thang đặc quyền
UCI	UCI Machine Learning Repository	Kho dữ liệu học máy UCI
UNSW	University of New South Wales	Đại học New South Wales

DANH MỤC CÁC BẢNG BIỂU

DANH MỤC CÁC HÌNH VẼ

PHẦN MỞ ĐẦU

1. Giới thiệu

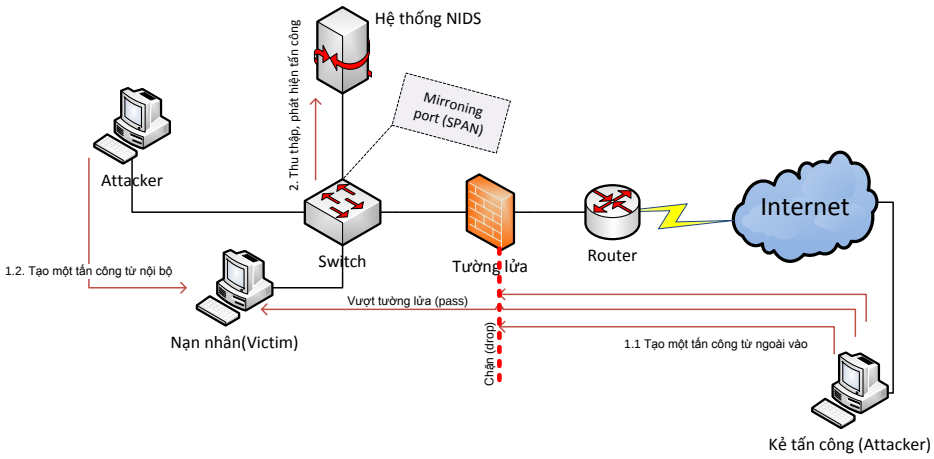
Cùng với sự phát triển nhanh chóng của hạ tầng, dịch vụ mạng máy tính và IoT (sau đây gọi tắt là mạng) đó là sự tăng nhanh của các loại hình tấn công mạng. Theo báo cáo thường niên có uy tín hàng đầu về mối đe doạ an ninh mạng trên toàn Thế giới năm 2018 và 2019 (có tên Internet Security Threat Report [1], viết tắt là ISTR). Số lượng mối đe doạ tấn công mạng tiếp tục tăng bùng nổ; khoảng 1/10 (một trong mười) các tên miền (URL) trên Internet là độc hại, số lượng tấn công Web tăng 56% trong năm 2018, số lượng thư rác (Spam) tăng khoảng 50% trong 4 năm liên tiếp từ 2015 đến 2018.

Các loại tấn công mới như Supply-Chain (một kiểu tội phạm mạng) tăng vọt 78%, mã độc PowerShell-Script tăng 1000%. Các loại tấn công này sử dụng kỹ thuật LolL (Living-off-the-land), kỹ thuật này cho phép các mã độc ẩn bên trong các gói tin nên khó bị phát hiện bởi các bộ dò tìm truyền thống. Số liệu cũng thể hiện các tấn công chủ yếu xuất phát từ một động lực rõ ràng ($\approx 85\%$), mục tiêu tập trung vào thu thập dữ liệu tình báo ($\approx 90\%$) [27].

Việc tìm giải pháp cho phát hiện và ngăn chặn các tấn công mạng đã thu hút sự quan tâm của rất nhiều nhà nghiên cứu trong nhiều thập kỷ qua. Điển hình trong lĩnh vực này là nghiên cứu hệ thống phát hiện xâm nhập mạng (Network Intrusion Detection Systems -NIDS). Các hệ thống NIDS được xem là lớp bảo vệ thứ hai sau tường lửa quy ước để phát hiện ra các xâm nhập, các mã độc và các hành vi xâm hại hệ thống mạng thông qua quan sát đặc tính lưu lượng

[1] https://www.broadcom.com/support/security-center/publications/threat-report, đây là báo cáo phân tích dữ liệu từ hệ thống giám sát an ninh mạng toàn cầu, được biết như là tổ chức dân sự lớn nhất thế giới về lĩnh vực tình báo mạng. Hệ thống thu thập từ 123 triệu bộ thu thập tấn công mạng, hàng ngày vô hiệu hoá khoảng 142 triệu mối đe doạ mạng. Hệ thống đang giám sát các hành vi đe doạ mạng trên 157 quốc gia.

mạng [12], [22]. NIDS thường được triển khai trên mạng để phát hiện các tấn công mạng từ các hướng (từ ngoài, từ trong mạng nội bộ) như Hình 1. Các

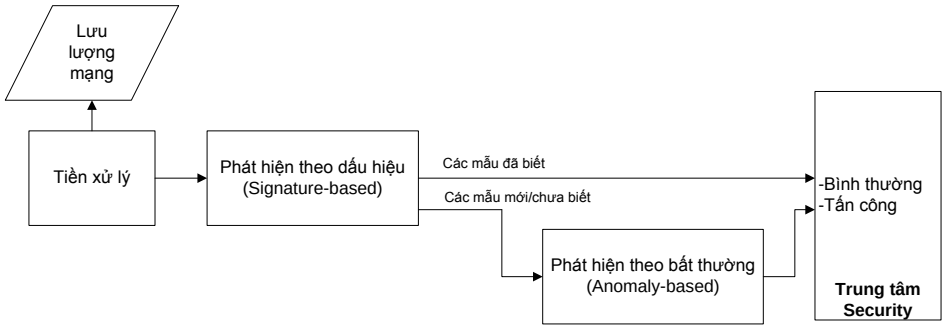


Hình 1: *Vị trí triển khai NIDS*

NIDS được chia thành hai loại: phát hiện dựa trên dấu hiệu (misuse-based hay signature-based) và phát hiện dựa trên sự bất thường (anomaly-based) [1], [2], [57], [82]. Việc phân nhóm căn cứ vào cách tiếp cận phát hiện xâm nhập. Các NIDS dựa trên dấu hiệu cho khả năng phát hiện chính xác các tấn công đã biết trước, trong khi đó chỉ có NIDS dựa trên hành vi bất thường mới có thể phát hiện được các tấn công mạng mới [45], [65], [116], nghiên cứu các phương pháp phát hiện bất thường (Anomaly Detection - AD) trong lĩnh vực an ninh mạng được biết đến với thuật ngữ là Network Anomaly Detection (NAD). Một hệ thống phát hiện xâm nhập hiệu quả thường được tạo thành từ giải pháp phát hiện dựa trên dấu hiệu và nối tiếp sau bởi giải pháp NAD [116] như Hình 2.

Bản chất nghiên cứu về NAD là nghiên cứu về bộ máy phát hiện (Detection Engine). Mô hình hoá hoạt động của bộ máy phát hiện bất thường để tìm kiếm giải pháp hiệu quả hơn trong phân tách các mẫu dữ liệu bình thường và bất thường.

Các phương pháp theo hướng cố gắng xác định độ lệch của dữ liệu đầu vào so với các mẫu dữ liệu sử dụng cho biểu diễn hoạt động thông thường của hệ



Hình 2: *Phương pháp signature-based nối tiếp bởi anomaly-based.*

thống đã được thiết lập trước, để đánh dấu các xâm nhập (các bất thường hay tấn công mạng). Do vậy, các giải pháp đề xuất cần quen với các mẫu sử dụng thông thường thông qua việc học [2]. Các phương pháp cho phép hệ thống "học" từ dữ liệu để giải quyết các bài toán cụ thể thường được biết đến với thuật ngữ học máy (machine learning). NAD là chủ đề nghiên cứu được đặc biệt quan tâm trong sự phát triển của lĩnh vực an ninh mạng [1], [20], đây là hướng đi cho tìm kiếm giải pháp phát hiện được các tấn công mới, chưa từng xuất hiện. Nhiều phương pháp học máy khác nhau đã được nghiên cứu, ứng dụng rộng rãi và đạt hiệu quả cao [45], [63].

Tuy nhiên, nghiên cứu NAD là để chuẩn bị tốt hơn cho các tấn công trong tương lai [63], đây là một chủ đề rộng và khó, với nhiều các thách thức như được trình bày trong phần tiếp theo.

2. Tính cấp thiết của luận án

Trong xây dựng các phương pháp phát hiện bất thường mạng, nhãn của tấn công được cho là không sẵn có trong quá trình huấn luyện mô hình [13], [20], [22]. Việc thu thập các tấn công gặp rất nhiều khó khăn do chúng thường được công bố không đầy đủ vì các cá nhân và tổ chức bị tấn công mạng muốn giữ bí mật nội bộ và bảo đảm quyền riêng tư [41], [91]. Việc gán nhãn cho một số lượng khổng lồ các hành vi bất thường mạng, qua đó đại diện cho toàn bộ các

bất thường trên hệ thống mạng là một nhiệm vụ tốn quá nhiều công sức và thời gian. Hơn nữa, các tấn công sau khi được nhận ra bởi các hệ thống phát hiện, thường cần một thời gian khá lớn để có thể xử lý và lấy mẫu. Trong khi các tấn công mới thường rất nguy hiểm đến hệ thống mạng. Đó là lý do NAD với mục tiêu chính là phát hiện ra các tấn công mới, cần phải thường xuyên được nghiên cứu, đổi mới. Hầu hết các nghiên cứu dựa trên tri thức đã biết đến về các tấn công thường không hiệu quả trong phát hiện các tấn công mới [5]. Do vậy quá trình huấn luyện các phương pháp NAD được khuyến nghị là hoàn toàn độc lập với dữ liệu tấn công, chỉ sử dụng dữ liệu bình thường cho xây dựng mô hình phát hiện bất thường [20].

Các kỹ thuật cho xây dựng các bộ phân lớp từ một lớp dữ liệu được gọi là phân đơn lớp (One-class classifications - OCC). Nhiều học giả đã đã chứng minh tính hiệu quả của phương pháp OCC cho NAD như có thể giải quyết được các vấn đề với không gian thuộc tính dữ liệu quá nhiều chiều (high-dimensional), có thể giúp ước lượng bộ siêu tham số (hyper-parameters) cũng như nâng cao khả năng phân lớp, giúp phát hiện ra các tấn công, mã độc mới (chưa từng biết) [20], [37], [110]. Các phương pháp OCC truyền thống có thể được chia thành các nhóm chính là: phương pháp dựa trên khoảng cách và phương pháp dựa trên mật độ [47]. Trong số đó, một số phương pháp nổi tiếng có thể giải quyết được các vấn đề của dữ liệu mạng như: Local Outlier Factor (LOF) [16] hoạt động hiệu quả trên dữ liệu không gian rất nhiều chiều; Kernel Density Estimation (KDE) [111] có thể tự học mà không cần giả định về phân bố của dữ liệu; One-Class Support Vector Machine (OCSVM) [88] hoạt động phù hợp cho nhiều lĩnh vực ứng dụng khác nhau. Gần đây, các phương pháp phát hiện bất thường dựa trên học sâu (deep learning) được cho là tiềm năng và hiệu quả hơn so với các phương pháp học máy truyền thống, nhất là trong điều kiện kích thước, số chiều dữ liệu quan sát ngày càng tăng nhanh [21]. Học sâu là thuật ngữ liên quan đến học cách biểu diễn dữ liệu (representation learning) với nhiều tầng, nhiều mức xử lý [66], là một nhánh của học máy. Học sâu được cho có

khả năng biểu diễn dữ liệu tốt hơn, cho phép tự học đặc tính dữ liệu (feature engineering) [20], [21], [86].

Trong số đó, các phương pháp học sâu dựa trên kiến trúc AutoEncoder (AE) được cho là kỹ thuật tiên tiến (the state-of-the-art) cho phát hiện bất thường mạng [20], [37], [100]. Để đáp ứng yêu cầu nâng cao khả năng phát hiện các tấn công mới và khó, việc nghiên cứu cải tiến phương pháp học sâu cho NAD phải luôn được quan tâm và là yêu cầu thiết thực. Shrink AE (SAE) [20], [37] được cho là phương pháp tiêu biểu gần đây cho phát hiện bất thường mạng phát triển dựa trên học sâu AutoEncoder. Phương pháp này được huấn luyện để tìm cách biểu diễn dữ liệu bình thường ở vùng rất chụm tại gốc toạ độ của không gian xem xét. Do vậy, với các đầu vào là dữ liệu bất thường (chưa từng biết đến), các vector ẩn tương ứng sẽ bị đẩy ra xa so với gốc toạ độ. Phương pháp dựa trên học sâu AutoEncoder này được cho là có khả năng phát hiện bất thường tốt hơn các phương pháp hiện thời trên nhiều tập dữ liệu kiểm thử phổ biến trong lĩnh vực học máy và an ninh mạng [20]. Tuy nhiên cơ chế hoạt động cũng cho thấy SAE vẫn cần được cải tiến, phát triển ở cả ở phần tiền xử lý dữ liệu trước SAE và lõi của SAE. Thứ nhất, vì mô hình học sâu này cố nén toàn bộ dữ liệu bình thường vào một cụm đơn duy nhất, do vậy thuật toán có thể không đạt hiệu quả tốt khi tập dữ liệu cho huấn luyện tồn tại ở dạng nhiều cụm (cluster). Thứ hai, mô hình SAE mặc dù cho khả năng phát hiện bất thường mạng rất tốt, tuy vậy SAE vẫn có thể gặp khó khăn với một số loại tấn công (bất thường). Đây là các mẫu tấn công khi được phân tách (kiểm tra) bởi SAE thường tạo ra các vector được biểu diễn ở gần gốc toạ độ hơn, do vậy việc phân tách giữa bình thường và bất thường khó hơn.

Theo cơ chế hoạt động của SAE, các tấn công mạng mà SAE gặp khó có thể do mẫu dữ liệu có nhiều điểm giống với mẫu dữ liệu bình thường, vì SAE cố ép để dữ liệu bình thường được biểu diễn ở vùng gần gốc toạ độ trong không gian biểu diễn mới. Do vậy với dữ liệu tấn công gần giống với dữ liệu bình thường cũng sẽ được biểu diễn gần tương tự, ở vùng rất gần nhau. Do vậy, với các mẫu

tấn công này, phương pháp NAD tiêu biểu dựa trên học sâu AutoEncoder này có thể không phân tách tốt giữa mẫu bình thường và bất thường.

Xác định ngưỡng ra quyết định là một bài toán khó khăn với các bộ phân đơn lớp OCC, đây là yêu cầu đối với mô hình khi triển khai trong thực tế [40]. Trong NAD, các mô hình dựa trên OCC khi thực thi cho đầu ra là độ đo mức độ bất thường (Anomaly Score - AS) của mẫu dữ liệu quan sát. Việc chỉ có một lớp dữ liệu cho huấn luyện, mô hình OCC thường cần phải sự can thiệp của chuyên gia trong xác định ngưỡng để phân tách bất thường và bình thường [21],[40].

Các phương pháp phát hiện xâm nhập đơn lẻ dù đã chứng minh rất hiệu quả, các phương pháp này được cho là thường chỉ hoạt động tốt với một loại tấn công mạng cụ thể [102], [117]. Điều này có thể giải thích như sau, các phương pháp (mô hình) được hình thành từ các thuật toán và dữ liệu [72], [112]. Do vậy cùng một thuật toán cụ thể, tính hiệu quả của phương pháp phụ thuộc vào dữ liệu được sử dụng cho huấn luyện mô hình. Các môi trường mạng khác nhau cho dữ liệu khác nhau, việc xử lý khác nhau cũng dẫn đến dữ liệu khác nhau và các tấn công mạng khác nhau cũng có dữ liệu khác nhau. Với sự phát triển nhanh, tinh vi của các loại tấn công mạng ngày nay kéo theo sự biến động và phức tạp của dữ liệu quan sát do vậy rất khó để một phương pháp đơn có thể đáp ứng khả năng phát hiện các xâm nhập, các bất thường. Trong trường hợp OCC, mỗi phương pháp đơn (Single AD - SlgAD) này biểu diễn dữ liệu lưu lượng mạng theo cách riêng của nó, do vậy độ lệch khi quan sát một mẫu dữ liệu đầu vào là rất khác nhau. Nói cách khác các phương pháp OCC thường có khả năng phát hiện bất thường rất khác nhau trong cùng một vấn đề đặt ra [21], [57], [69]. Theo Bhattacharyya [12], mỗi phương pháp đơn NAD có mức độ phụ thuộc vào môi trường ứng dụng khác nhau, do vậy sự cần thiết trong nghiên cứu đưa ra giải pháp hiệu quả trên nhiều môi trường mạng khác nhau. Ví dụ phương pháp KDE rất hiệu quả trong phát hiện các bất thường về thư rác nhưng lại không hiệu quả trong phát hiện các quảng cáo rác từ Internet. Ở chiều ngược lại, LOF rất hiệu quả trong phát hiện quảng cáo rác nhưng lại không hiệu quả trong phát

hiện bất thường là các thư rác. Do vậy làm thế nào để gom được lợi thế từ các phương pháp đơn OCC khác nhau là một yêu cầu rất thiết thực cần có lời giải.

Vấn đề kết hợp các ưu điểm từ các phương pháp đơn được huấn luyện bằng học có giám sát để tạo một bộ phát hiện có khả năng mạnh hơn đã được nhiều nghiên cứu thực hiện [68], [82], [102], [117]. Trong đó, Data Fusion (DF) [10], [68], [117], tạm dịch là tổng hợp dữ liệu, trong phạm vi luận án có nghĩa là tổng hợp quyết định từ đa máy phát hiện NAD, là giải pháp được nhiều học giả quan tâm cho kết hợp lợi thế của các phương pháp đơn, kỹ thuật đơn. Tuy vậy, rất nhiều các vấn đề khó khăn khi xây dựng một mô hình DF như sau. Đầu tiên là vấn đề mức DF, cơ bản có ba mức hoạt động: mức dữ liệu (data fusion layer), mức thuộc tính (feature fusion layer) và mức quyết định (decision fusion layer) [68], [102], [105]. Vấn đề thứ hai cần quan tâm khi phát triển phương pháp DF là xác định cơ sở để lựa chọn các phương pháp đơn nhằm giúp cho phương pháp DF đạt hiệu quả cao. Vấn đề thứ ba là xác định thuật toán sử dụng cho DF, đây được xem là mấu chốt cho một hệ thống DF và thường phụ thuộc yêu cầu của ứng dụng cụ thể. Các nghiên cứu gần đây [68], [69], [82], [92], [104] cho thấy lý thuyết dựa trên dẫn chứng (Evidence Theory hay Dempster-Shafer Theory) là giải pháp tiềm năng cho xây dựng mô hình phát hiện xâm nhập theo hướng DF. Thuận lợi của lý thuyết Dempster-Shafer (D-S) nằm ở điểm lý thuyết này không yêu cầu xác suất tiền nghiệm (tiền tri thức) như phương pháp suy luận ra quyết định nổi tiếng Bayes, do vậy ứng dụng D-S được xem là tiềm năng cho các bài toán phát hiện bất thường [25].

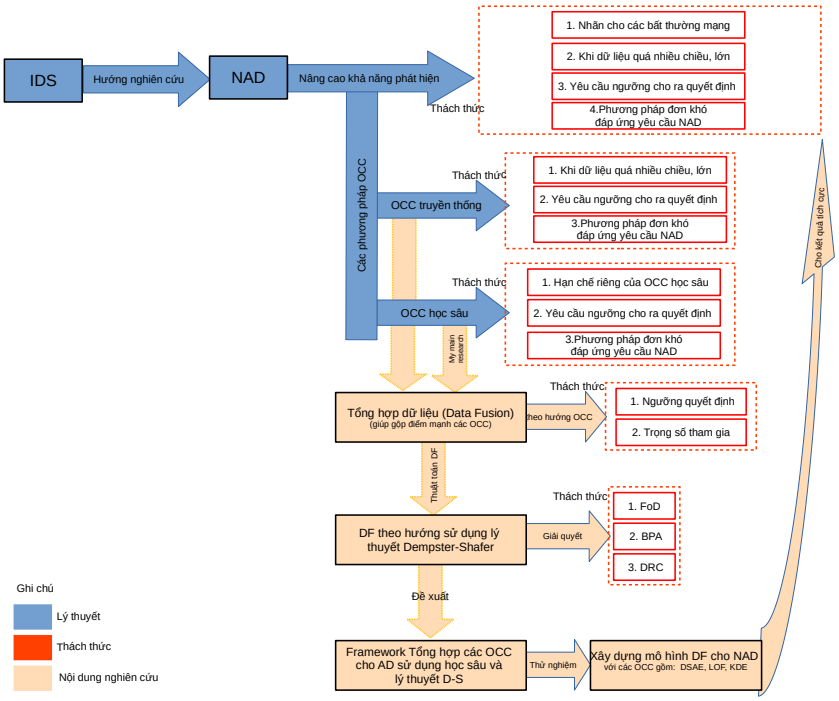
Phương pháp DF đã được sử dụng nhiều để tạo ra một mô hình đồng nhất NAD qua việc tổng hợp tri thức từ các bộ phân lớp đơn được huấn luyện có giám sát [68], [102]. Việc tổng hợp được tạo ra dựa trên ngưỡng của các bộ phân lớp và trọng số của các bộ phân lớp cục bộ. Tuy nhiên khi áp dụng DF cho bài toán các bộ phân lớp cục bộ OCC thì gặp rất nhiều thách thức như làm thế nào để xác định ngưỡng, cơ sở cho việc xác định trọng số niềm tin của các thành viên tham gia tổng hợp, cụ thể:

- Vấn đề ngưỡng quyết định của các phương pháp đơn khi tham gia DF: Đó là không có cơ sở để xác định ngưỡng cho các phương pháp OCC cục bộ, điều này được giải thích vì các OCC chỉ sử dụng duy nhất dữ liệu bình thường cho huấn luyện mô hình. Thường không có dữ liệu bất thường để ước lượng ngưỡng cho độ đo bất thường. Trong trường hợp nếu có thêm ít dữ liệu bất thường cho việc ước lượng thì vô hình dung lại tác động xấu đến khả năng phân lớp của mô hình [20]. Do vậy, việc ướng lượng ngưỡng bất thường cho các bộ phân lớp đơn khác nhau khi tham gia tổng hợp là công việc khó trong xây dựng mô hình DF chỉ từ các phương pháp OCC.

- Tiếp đến là vấn đề trọng số của các phương pháp đơn khi tham gia DF: Giả sử có thể giải quyết vấn đề ngưỡng cho các OCC cục bộ khi tham gia mô hình DF, vậy làm thế nào để xác định trọng số cho các phân lớp đơn khi tham gia. Thực tế, một số bộ phân lớp đơn có độ tin cậy cao hơn khi tham gia mô hình DF, tuy nhiên vì chỉ có dữ liệu bình thường cho huấn luyện các phương pháp đơn, do vậy không có cơ sở để ước lượng trọng số như thường được thực hiện trong phương pháp học có giám sát. Đây là một thách thức cần phải giải quyết để đạt mục tiêu xây dựng một mô hình DF.

Theo như NCS được biết, chưa có nghiên cứu về xây dựng mô hình DF từ các phương pháp OCC và lý thuyết D-S để nâng cao khả năng phát hiện bất thường mạng. Ngoài các thách thức khi xây dựng mô hình DF cho OCC, nhiều thách thức đặt ra cần phải giải quyết khi áp dụng lý thuyết D-S như: xây dựng hàm gán niềm tin cơ sở BPA (Basic Probability Assignment); giải pháp áp dụng hàm kết hợp DRC (D-S Rule Combination), vì hàm này xem các nguồn cung cấp thông tin có độ tin cậy như nhau, điều này được cho là hạn chế vì không phù hợp thực tế [69], [73], [92].

Hình 3 minh họa hướng nghiên cứu của luận án, các đối tượng màu xanh thể hiện thực trạng cơ sở lý luận liên quan đến nghiên cứu về NAD, màu đỏ thể hiện các thách thức theo mỗi hướng nghiên cứu gặp phải, còn màu vàng thể hiện



Hình 3: *Sơ đồ trình bày hướng nghiên cứu của luận án*

hướng nghiên cứu được chọn cho luận án này.

3. Phát biểu bài toán

Từ tính cấp thiết của luận án như đã phân tích trên là động lực thúc đẩy để NCS hướng đến cải tiến, phát triển phương pháp phát hiện bất thường mạng. Việc phát triển mô hình NAD cần giải quyết các thách thức đối với mô hình NAD tiêu biểu dựa trên học sâu; phát triển mô hình khung cho NAD dựa trên tổng hợp dữ liệu. Chi tiết về các phát biểu bài toán gồm:

- Vấn đề thứ nhất, phương pháp học sâu dựa trên AutoEncoder được cho là phương pháp tiên tiến cho phát hiện bất thường mạng. Do vậy, nhiệm vụ nghiên cứu cải tiến NAD cần phải tiếp tục phát triển phương pháp tiêu biểu dựa trên học sâu để ngày càng đáp ứng tốt hơn yêu cầu thực tiễn, khi mà các tấn công (bất thường) mạng luôn luôn thay đổi.

- Thứ hai, tổng hợp dữ liệu theo hướng lý thuyết D-S được cho là giải pháp tiềm năng để có thể gom được các lợi thế từ các phương pháp đơn. Do vậy, nghiên cứu cải tiến NAD cần phải đưa ra giải pháp mang tính khung cho việc tổng hợp được lợi thế từ các phương pháp đơn OCC và có thể áp dụng hiệu quả cho lĩnh vực an ninh mạng.

- Thứ ba, nghiên cứu phát triển mô hình NAD cần phải đưa ra giải pháp tự động thiết lập ngưỡng ra quyết định. Theo đó, giải pháp đề xuất có thể hoạt động được trên môi trường thực tế mà không cần sự hỗ trợ của chuyên gia trong việc thiết lập ngưỡng.

4. Mục tiêu của luận án

Mục tiêu chính của luận án là đóng góp khoa học cho lĩnh vực nghiên cứu phát hiện bất thường thông qua việc đề xuất các giải pháp có thể giải quyết một số các thách thức mà các mô hình tiêu biểu trong lĩnh vực NAD đang gặp phải. Để đạt mục tiêu tổng quát này, một số mục tiêu cụ thể như sau:

- Phát triển phương pháp học sâu cho NAD theo hướng cải tiến mô hình học sâu tiêu biểu hiện có, cụ thể là phương pháp học sâu dựa trên AutoEncoder. Một số hạn chế của phương pháp này cần phải được nghiên cứu và cải tiến như đã được đề cập ở phần đặt vấn đề của luận án.

- Phát triển được mô hình khung của NAD dựa trên tổng hợp dữ liệu sử dụng lý thuyết D-S, mô hình kết hợp được lợi thế từ các phương pháp đơn OCC dựa trên cả học sâu và truyền thống. Thêm vào đó, mô hình đề xuất cần có khả năng tự ước lượng ngưỡng quyết định, giúp giải pháp phù hợp với yêu cầu thực tế, không cần sự can thiệp của chuyên gia trong xác định ngưỡng.

5. Đối tượng và Phạm vi luận án

- Đối tượng nghiên cứu của luận án là các phương pháp phát hiện bất thường, mô hình tổng quan và các hướng kỹ thuật được sử dụng để nâng cao khả năng phát hiện bất thường mạng.

- Phạm vi luận án là lĩnh vực phát hiện bất thường mạng (Network Anomaly Detection), các kỹ thuật học sâu (Deep learning), tổng hợp dữ liệu (Data Fusion - DF) và lý thuyết Dempster-Shafer (D-S). Các vấn đề của luận án đều trên giả định chỉ có dữ liệu bình thường trong quá trình huấn luyện các mô hình phát hiện bất thường. Luận án sử dụng nhiều bộ dữ liệu phổ biến để phục vụ cho kiểm thử kết quả lý thuyết. Các bộ dữ liệu này được sử dụng rộng rãi trong các công trình nghiên cứu liên quan và sẵn tại các nguồn chính thống, trên mạng Internet.

6. Phương pháp nghiên cứu

Luận án sử dụng phương pháp nghiên cứu tổng hợp, phân tích. Khảo sát tổng quan các kết quả nghiên cứu trong và ngoài nước gần đây liên quan đến phát hiện bất thường. Phân tích các vấn đề còn hạn chế, các hướng nghiên cứu được gợi ý để từ đó đề xuất hướng đi cụ thể, từ đó tiến hành khảo sát chuyên sâu các bài toán đặt ra.

Sử dụng các công cụ toán học, các lý thuyết để đề xuất các mô hình phát hiện bất thường mạng theo hướng giải quyết bài toán đặt ra. Sử dụng các bộ dữ liệu phổ biến trong lĩnh vực an ninh mạng, tiến hành cài đặt mô hình để kiểm chứng kết quả nghiên cứu lý thuyết.

7. Đóng góp của luận án

- Luận án đã đề xuất cải tiến mô hình tiêu biểu phát hiện bất thường dựa trên học sâu, giải pháp đề xuất được trình bày gồm hai thành phần, trình bày thông qua hai mô hình KSAE (Clustering-based Shrink AutoEncoder) và DSAE (Double-Shrink AutoEncoder).
- Luận án đề xuất được một phương pháp khung phát hiện bất thường dựa trên tổng hợp dữ liệu có tên OFuseAD (One-class Fusion-based Anomaly Detection). Thực nghiêm cho thấy, OFuseAD hoạt động khả thi, đạt hiệu quả và độ ổn định; ngoài ra mô hình có khả năng tự động đưa ra ngưỡng quyết định.

7. Bố cục luận án

- Phần mở đầu trình bày về vấn đề an ninh mạng, tính cấp thiết của luận án, mục tiêu phạm vi, phương pháp và những đóng góp chính của luận án.
- Chương 1 trình bày các kiến thức cơ sở và liên quan đến phát hiện bất thường mạng.
- Chương 2 trình bày kết quả nghiên cứu phát triển một số mô hình NAD dựa trên học sâu, tập trung vào giới hai mô hình NAD mới là KSAE và DSAE để khắc phục hạn chế của SAE, là mô hình NAD tiêu biểu dựa trên học sâu.
- Chương 3 trình bày kết quả nghiên cứu mô hình NAD dựa trên tổng hợp dữ liệu, tập trung giới thiệu mô hình OFuseAD, sử dụng lý thuyết D-S để gom lợi thế từ các phương pháp đơn OCC gồm cả học sâu và truyền thống.
- Cuối cùng, Phần kết luận trình bày tóm lược các nội dung chính, các kết quả chủ yếu của luận án, một số hạn chế cũng như một số hướng nghiên cứu phát triển trong tương lai.

CHƯƠNG 1. TỔNG QUAN VỀ PHÁT HIỆN BẤT THƯỜNG MẠNG

Chương này trình bày một số kiến thức cơ sở, các nghiên cứu liên quan về phát hiện bất thường mạng, nội dung gồm bốn phần chính. Phần thứ nhất làm rõ khái niệm về phát hiện bất thường mạng, mô hình tổng quan, các thành phần. Phần thứ hai trình bày một số kết quả nghiên cứu liên quan, gồm cả phương pháp truyền thống và phương pháp học sâu. Tiếp đó, trình bày phương pháp kết hợp, tổng hợp và một số kết quả nghiên cứu liên quan đến tổng hợp dữ liệu cho xây dựng mô hình phát hiện xâm nhập mạng, giới thiệu lý thuyết Dempster - Shafer (D-S). Mục đích của phần này giúp làm rõ hơn về hướng nghiên cứu.

Trong phần còn lại, luận án giới thiệu một số bộ dữ liệu cho kiểm thử, các chỉ số đánh giá hiệu quả của các mô hình phát hiện bất thường. Một số kết quả nghiên cứu đã được công bố trên công trình [CT4] (trong phần CÁC CÔNG TRÌNH CÓ LIÊN QUAN ĐẾN LUẬN ÁN)

1.1. Hệ thống phát hiện bất thường mạng

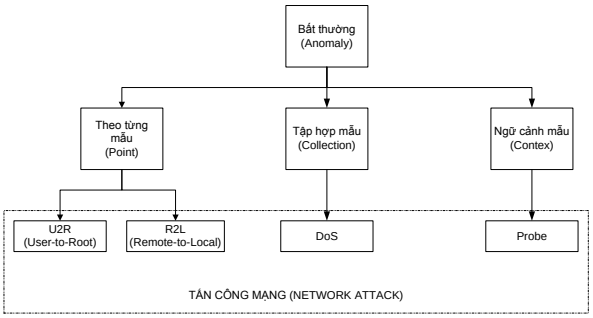
1.1.1. Khái niệm

Phát hiện bất thường (Anomaly Detection - AD) là việc tìm ra các mẫu dữ liệu có sự khác biệt so với các mẫu dữ liệu còn lại, các mẫu dữ liệu được phân biệt này thường được gọi là bất thường (anomaly) [13], [22], [86]. Nguyên nhân của các mẫu bất thường này thường từ các vấn đề mới hoặc chưa từng được biết đến của đối tượng mà hệ thống đang quan sát, xử lý.

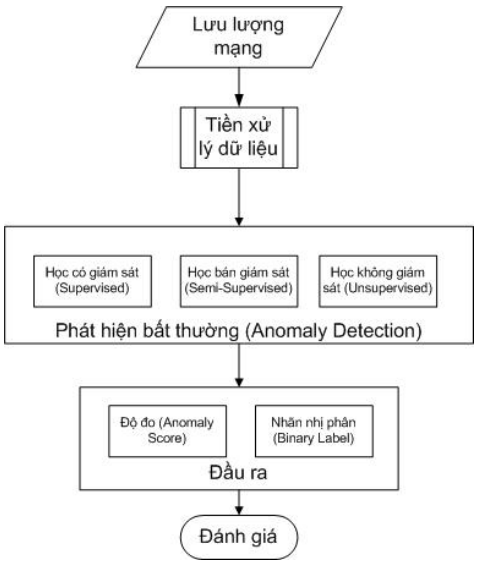
AD là một mảng nghiên cứu rộng, được rất nhiều các học giả quan tâm và được ứng dụng trong rất nhiều các lĩnh vực [21]. Các nghiên cứu quan tâm đến

cải tiến khả năng phát hiện bất thường sử dụng các kỹ thuật thống kê và học máy [22]. Một số các thuật ngữ khác của AD được biết đến như phát hiện cái mới (novelty detection, phát hiện cái sai lệch (deviation detection) [4], [5]. Mặc dù có rất nhiều các định nghĩa khác nhau, một trong số đó được chấp nhận rộng rãi như định nghĩa của Hawkins [51]: "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".

Trong lĩnh vực an ninh mạng, AD được biết đến với thuật ngữ phát hiện bất thường mạng (Network Anomaly Detection - NAD), ngoài ra các thuật ngữ khác như Network anomaly – Based IDS, Network profile-based IDS, Network Novelty Detection cũng được sử dụng. NAD là các kỹ thuật tìm ra các mẫu dữ liệu bất thường trong lưu lượng mạng mà nó không giống với mẫu dữ liệu được cấu thành từ các hoạt động bình thường của mạng [5], [13], [39]. Các bất thường có thể đến từ các tấn công mạng, lỗi trong cấu hình hệ thống mạng hoặc là do các hành vi của người dùng sai với chính sách an toàn mạng. Có thể phân thành ba trường hợp nhận biết bất thường [5], [21]: (1) Theo từng mẫu (Point), nghĩa là chỉ một điểm dữ liệu cũng thể hiện được sự bất thường; (2) Tập hợp mẫu (Collection), là sự bất thường được thể hiện từ một tập hợp các mẫu dữ liệu; (3) Ngữ cảnh mẫu (Context), trong một ngữ cảnh cụ thể thì các điểm dữ liệu mới thể hiện sự bất thường. Trong phạm vi luận án, từ bất thường được xem là các tấn công, các hành vi phá hoại mạng. Hình 1.1 đưa ra ánh xạ giữa loại bất thường và nhóm tấn công mạng [5]. Trong đó, DoS là các tấn công từ chối dịch vụ, với một yêu cầu đến máy chủ dịch vụ là bình thường, tuy nhiên một "tập hợp mẫu" rất lớn các yêu cầu thường là dạng tấng công DoS. Proble là loại tấn công dò quét, được cho là bất thường theo "ngữ cảnh mẫu" vì loại tấn công này là các truy vấn để hỏi/đáp nhằm thu thập thông tin trong các điều kiện hiện trạng hạ tầng và dịch vụ mạng vẫn không có gì thay đổi. U2R là loại tấn công leo thang đặc quyền, và R2L là các loại tấn công chiếm quyền máy tính cục bộ. Đây là các loại tấn công tinh vi và được cho là bất thường ở từng mẫu [4], [5].



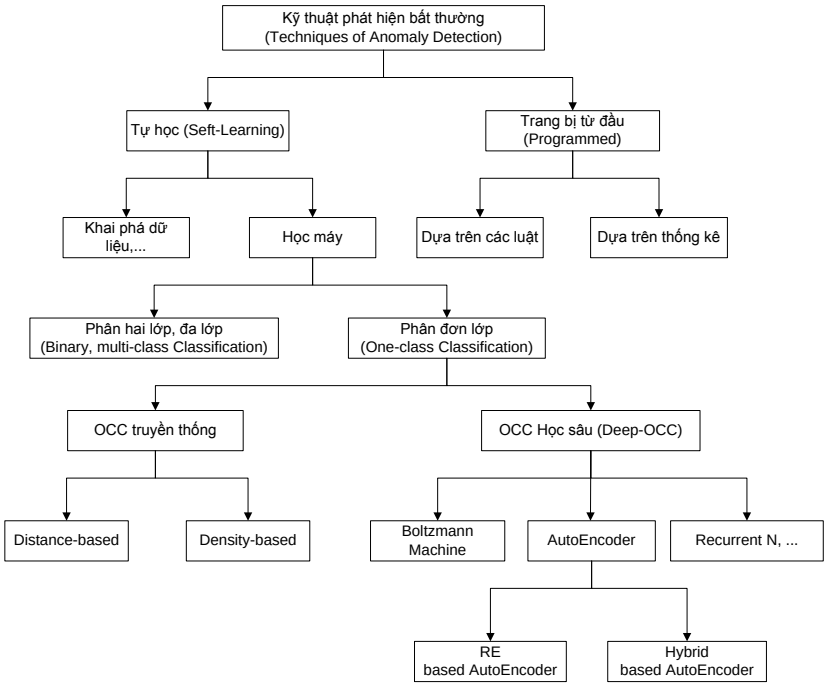
Hình 1.1: *Nhóm tấn công mạng và loại bất thường, Hình từ Ahmed [2016] [5]*



Hình 1.2: *Kiến trúc chung của NAD, Hình từ Ahmed [2016] [5]*

1.1.2. Mô hình phát hiện bất thường mạng

Kiến trúc tổng thể của mô hình phát hiện bất thường mạng có thể được mô tả như Hình 1.2. Theo đó, lưu lượng mạng sau khi được thu thập, xử lý và trích chọn đặc trưng sẽ được thực hiện tiền xử lý. Việc tiền xử lý dữ liệu thường chủ yếu là thực hiện các chuyển đổi về định dạng (như biểu diễn dạng mã hoá nhị phân hoá), co dãn dữ liệu (như co dữ liệu về [-1,1]), lọc đặc trưng hay xoá bỏ các dữ liệu ngoại lai. Trong phạm vi luận án, việc tiền xử lý dữ liệu chỉ đơn giản thực hiện việc mã hoá nhị phân hoá và co dãn dữ liệu, được thực hiện khi thực nghiệm như các nghiên cứu khác đã làm [18], [19], [20]. Dữ liệu sau khi tiền xử lý sẽ được đưa vào máy phát hiện (bộ phân lớp) bất thường, đây là thành phần



Hình 1.3: *Sơ đồ phân loại các kỹ thuật phát hiện bất thường [21], [63]*

chính của hệ thống NAD. Nhìn chung [21], các kỹ thuật dùng cho bộ phân lớp NAD có thể được phân loại theo như Hình 1.3. Trong số đó, học máy (machine learning) đang được cho là lựa chọn chính của các học giả khi nghiên cứu bất thường trong lĩnh vực an ninh mạng [5], [21], [43]. Học máy là phương pháp giúp cho máy có được tri thức để cung cấp dữ liệu đầu ra theo mục tiêu nó được huấn luyện, mong muốn đặt ra. Các thuật toán học máy thường được chia thành ba nhóm chính: học có giám sát; học không giám sát và học bán giám sát. Trong bài toán NAD có thể được trình bày như sau:

- Phát hiện thông qua học có giám sát (Supervised Anomaly Detection): Học giám sát yêu cầu phải có nhãn của cả dữ liệu bình thường và bất thường trong quá trình huấn luyện mô hình phát hiện bất thường. Nhìn chung, các phương pháp học giám sát cho kết quả phát hiện tốt hơn so với các phương pháp còn lại, vì phương pháp này có nhiều thông tin về dữ liệu cần phân tích hơn. Tuy nhiên, trong phạm vi phát hiện bất thường, để có thể đạt được hiệu quả phát hiện tốt hơn, dữ liệu huấn luyện cần phải có đủ (càng

nhiều càng tốt) số nhãn của dữ liệu bình thường và từng loại bất thường. Tuy nhiên việc thu thập và gán các nhãn này là nhiệm vụ vô cùng khó khăn và thách thức [21], [41], [91], [113], vì vậy để phát hiện các tấn công mới, phương pháp học giám sát không mạnh hơn các phương pháp khác. Có nhiều thuật toán điển hình cho phương pháp học giám sát như mạng nơ-ron nhân tạo (Artificial Neural Networks), máy vector hỗ trợ (Support Vector Machines - SVM), k láng giềng gần nhất (k- Nearest Neighbors), mạng Bayes (Bayesian Networks), và cây quyết định (Decision Trees).

- Phát hiện thông qua học không giám sát (Unsupervised Anomaly Detection): Học không giám sát không cần nhãn cho quá trình huấn luyện mô hình, cơ chế hoạt động của học không giám sát trong xây dựng mô hình NAD dựa trên các giả định sau [17], [81]. Thứ nhất, dữ liệu lưu lượng mạng là bình thường và chỉ một phần rất bé có thể là dữ liệu tấn công. Thứ hai, xét về mặt thống kê, dữ liệu lưu lượng mạng tấn công và bình thường là khác nhau. Thêm đó, không cần bất cứ nhãn nào cho quá trình huấn luyện mô hình. Dựa trên các giả định đó, mô hình phát hiện bất thường được huấn luyện để tách biệt hai nhóm. Nhóm bình thường là các trường hợp có tính tương tự nhau và xuất hiện thường xuyên. Nhóm bất thường là các trường hợp xuất hiện không thường xuyên và rất khác so với phần đa các trường hợp hiện có [22]. Trong phạm vi luận án không áp dụng các giả định này, do vậy phương pháp học không giám sát không được sử dụng trong suốt nghiên cứu của luận án. Điển hình cho học không giám sát là các các thuật toán phân cụm (như K-means [67]).

- Phát hiện thông qua học bán giám sát (Semi-supervised Anomaly Detection): Phương pháp học bán giám sát được coi là nằm giữa học có giám sát (yêu cầu có càng nhiều càng tốt số nhãn khi huấn luyện) và học không giám sát (không cần nhãn khi huấn luyện). Phương pháp này thường sử dụng dữ liệu không có nhãn kết hợp với một số lượng nhỏ dữ liệu được gắn nhãn. Do

vậy giúp giảm thiểu rất lớn công sức gán nhãn dữ liệu trong khi vẫn có thể đạt hiệu quả phát hiện bất thường tương đồng như học có giám sát [17]. Khi áp dụng phương pháp học bán giám sát cho lĩnh vực an ninh mạng, chúng ta giả định rằng chỉ có dữ bình thường được gắn nhãn để huấn luyện cho mô hình. Điều này cũng thực tế hơn việc áp dụng học có giám sát vì không đặt ra yêu cầu cần phải gán nhãn cho bất cứ dữ liệu bất thường nào. Mặc dù vẫn có một số nghiên cứu NAD theo hướng học bán giám sát với giả định có một số lượng nhất định dữ liệu bất thường cho quá trình huấn luyện mô hình [29], [30]. Tuy nhiên các kết quả này thường không được áp dụng rộng rãi vì việc thu thập, gán đủ nhãn cho bất thường để đại diện được cho toàn bộ trường hợp bất thường của hệ thống là điều không thể. Do vậy, quá trình huấn luyện mô hình NAD được khuyến nghị là hoàn toàn độc lập với việc sẵn có của dữ liệu bất thường. Những vấn đề đặt ra trên là lý do mà luận án chọn phát triển các thuật toán phát hiện bất thường mạng theo hướng học bán giám sát. Một vài thuật toán phổ biến sử dụng học bán giám sát như LOF [16], KDE [111], OCSVM [88] và SAE [20].

Do vậy, mô hình học máy được huấn luyện theo học bán giám sát là phù hợp cho xây dựng máy phát hiện NAD, là các bộ phân đơn lớp OCC, gọi chung là mô hình NAD. Quá trình kiểm thử, độ lệch nhau trên không gian biểu diễn mới giữa mẫu dữ liệu đầu vào và dữ liệu đã được huấn luyện được sử dụng làm cơ sở để phân tách bất thường và bình thường. Trong nội dung tiếp theo sẽ trình bày về vấn đề cốt lõi của thành phần đầu vào, thành phần đầu ra của mô hình tổng quan NAD.

1.1.3. Lưu lượng mạng

Đầu vào của mô hình NAD cơ bản là lưu lượng mạng, dữ liệu lưu lượng mạng được thu thập bằng các công cụ chặn, bắt (gọi là sniffer), tập dữ liệu thô này gồm các gói tin được cấu trúc (ví dụ theo bộ giao thức TCP/IP đối với mạng máy tính). Một số các bộ phát hiện xâm nhập (như Snort [85], là một

signatured-based NIDS) sử dụng trực tiếp các gói tin để phát hiện xâm nhập. Tuy nhiên, nhiều tấn công chỉ có thể nhận ra khi quan sát dữ liệu ở mức phiên (session hay flow), do vậy dữ liệu mạng thô thu thập được thường được xử lý để trích chọn đặc trưng (Feature selection) ở cả mức gói tin và mức phiên công tác. Việc trích chọn đặc trưng được thực hiện bởi các thuật toán khác nhau [105]. Các thuộc tính cơ bản được chia làm hai nhóm: số (numerical) và tập hợp (catagorical). Nhóm dữ liệu số gồm hai nhóm con là: rời rạc (discrete data), để biểu diễn các thuộc tính có tính đếm được; liên tục (continuous data), để biểu diễn các thuộc tính chỉ có thể biểu diễn bởi số thực. Việc trích chọn đặc trưng có ý nghĩa hết sức quan trọng trong lĩnh vực phát hiện bất thường [12], [13], [105], việc giảm số chiều dữ liệu sẽ tăng hiệu năng thuật toán, tăng chất lượng thuộc tính sẽ tăng hiệu quả thuật toán, tăng tỉ lệ báo cảnh đúng và giúp cho việc biểu diễn dữ liệu được tường minh hơn. Các thuộc tính lưu lượng mạng được tính toán trên cơ sở giá trị tương ứng trong gói tin và phiên kết nối. Trong mạng máy tính, các thuộc tính được chia làm 03 nhóm: 1) Basic features: Bao gồm các thuộc tính có thể thu thập được từ một phiên kết nối TCP/IP. 2) Traffic features: Là các thuộc tính được tính dựa trên giá trị trường window trong gói tin TCP/IP. 3) Content features: Các thuộc tính được trích chọn từ phần nội dung (content) của TCP/IP.

Trong lĩnh vực phát hiện xâm nhập mạng, các bộ dữ liệu (datasets) được tạo thành từ lưu lượng mạng, theo các phương pháp trích chọn đặc trưng khác nhau, để phục vụ đánh giá độ tin cậy của các giải pháp an ninh mạng, các bộ dữ liệu sử dụng trong phạm vi luận án được trình bày tại phần 1.4.

1.1.4. Đầu ra của mô hình NAD

Mô hình NAD thường cho đầu ra như mô tả trên Hình 1.2, có hai dạng đầu ra cho mô hình là: độ đo bất thường; và nhãn nhị phân. Trong đó, các mô hình phát hiện bất thường hướng đến mục tiêu cho đầu ra là nhãn nhị phân, vì nếu đầu ra là độ đo bất thường thì mô hình vẫn cần tiếp tục có sự hỗ trợ của chuyên

gia trong việc định ngưỡng [40], [74].

- Độ đo bất thường (Anomaly score - AS): Theo loại đầu ra này, mô hình dự đoán sẽ cung cấp một xác suất ứng với mỗi điểm dữ liệu đầu vào, được gọi là độ đo bất thường có giá trị trong khoảng (0,1). Độ đo này chỉ ra mức độ bất thường xét cho điểm dữ liệu đầu vào. Tuy vậy vấn đề lớn nhất khi sử dụng loại đầu ra này là hệ thống vẫn cần thêm ngưỡng quyết định (Decision Threshold) để xác định điểm dữ liệu bình thường hay không. Trong phạm vi luận án này, một số kết quả nghiên cứu vẫn sử dụng AS cho đánh giá mô hình, cụ thể là khi phát triển các mô hình NAD sử dụng mạng nơ-ron học sâu như được trình bày tại Chương 2.
- Nhãn nhị phân (Binary Label - BL): Các mô hình cho dữ liệu đầu ra loại này thường gắn 1 cho trạng thái bất thường và 0 cho trạng thái bình thường của hệ thống mạng đang giám sát. Mô hình cho đầu ra dạng này có thể coi là cung cấp tri thức phù hợp với bài toán phát hiện các đối tượng mới, chưa nhìn thấy bao giờ, là bất thường hay bình thường; mô hình với đầu ra BL cũng được coi là cung cấp thông tin cụ thể hơn so với dạng có đầu ra AS. Về cơ bản, một hệ thống thực yêu cầu phải chỉ rõ có bất thường hay không chứ không dừng lại ở một độ đo bất thường [20], [40]. Luận án cũng tiến tới mục tiêu cung cấp thông tin đầu ra (tri thức) ở mức nhãn nhị phân (BL), trình bày cụ thể về kết quả này thể hiện tại Chương 3.

1.2. Một số phương pháp đơn cho phát hiện bất thường mạng

Các phương pháp phát hiện bất thường chủ yếu dựa trên thống kê, khai phá dữ liệu và học máy [1]. Việc phân loại các kỹ thuật có nhiều quan điểm khác nhau và các thuật toán cho AD thường có những phần chồng lấn [5]. Hình 1.3 trình bày một cách phân loại các kỹ thuật sử dụng cho NAD, các kỹ thuật này được phân thành hai nhóm chính là có khả năng tự học (self-learning) hay được

lập trình (trang bị kiến thức) rõ từ đầu [63]. Trong số đó, các kỹ thuật phát hiện bất thường dựa trên học máy theo hướng phân đơn lớp OCC được đánh giá là phù hợp và tiềm năng cho lĩnh vực an ninh mạng [19], [20]. Điều này vì các mô hình NAD được cho là phù hợp, có tiềm năng hơn khi chỉ sử dụng mỗi dữ liệu bình thường cho huấn luyện, như đã trình bày ở Phần mở đầu.

Các phương pháp OCC được cho là có thể giải quyết được các vấn đề với không gian thuộc tính dữ liệu quá nhiều chiều (high-dimensional), có thể giúp ước lượng bộ siêu tham số (hyper-parameters) cũng như nâng cao khả năng phân lớp, giúp phát hiện ra các tấn công, mã độc mới (chưa từng biết) [20], [37], [110]. Các phương pháp OCC có thể được phân thành hai nhóm, phương pháp OCC truyền thống và phương pháp OCC học sâu, nội dung trình bày sau đây sẽ giới thiệu các phương pháp OCC được cho là phổ biến, được nhiều nghiên cứu về NAD sử dụng trong những năm gần đây. Các thuật toán OCC giới thiệu trong phần này cũng được sử dụng cho các thử nghiệm liên quan trong suốt luận án. Trong các mô hình phát hiện bất thường mạng, các thuật toán trên có thể đóng vai trò như là các phương pháp độc lập, thực thi từ nguyên bản dữ liệu thuộc tính đầu vào hay được đặt phía sau một phương pháp giảm chiều dữ liệu (feature reduction).

1.2.1. Một số phương pháp OCC truyền thống

Các phương pháp OCC truyền thống đã chứng minh rất hiệu quả trong lĩnh vực NAD, trong số đó, một số phương pháp nổi tiếng có thể giải quyết được các vấn đề của dữ liệu mạng như: Local Outlier Factor (LOF) [16] hoạt động hiệu quả trên dữ liệu không gian rất nhiều chiều; Kernel Density Estimation (KDE) [111] có thể tự học mà không cần giả định về phân bố của dữ liệu; One-Class Support Vector Machine (OCSVM) [88] hoạt động phù hợp cho nhiều lĩnh vực ứng dụng khác nhau.

Các phương pháp OCC truyền thống có thể được chia thành các nhóm chính là: phương pháp dựa trên khoảng cách và phương pháp dựa trên mật độ [47].

Ngoài ra, các phương pháp dựa trên vector hỗ trợ có thể được xem là phổ biến và nổi tiếng nhất, phương pháp Centroid (CEN) đơn giản, dễ cài đặt và không cần tham số. Trong phần tiếp theo, sẽ trình bày lần lượt các phương pháp trên.

1.2.1.1. Phương pháp OCC dựa trên khoảng cách

Phương pháp phát hiện bất thường dựa trên khoảng cách (distance-based) thường sử dụng phương pháp tính khoảng cách Euclid giữa các điểm dữ liệu. Về cơ bản, các thuật toán này tạo ra một độ đo bất thường được tính toán dựa trên khoảng cách tương quan giữa các điểm dữ liệu. Các điểm cho độ đo lớn hơn ở một mức độ tương đối, sẽ được xem là bất thường [72], [78]. Phổ biến là các phương pháp xem xét khoảng cách giữa một điểm dữ liệu với các láng giềng của nó, kỹ thuật được biết đến là láng giềng gần nhất (nearest-neighbor). Phương pháp này hoạt động trên giả định rằng, các điểm dữ liệu bình thường thường nằm rất sát nhau, còn các điểm dữ liệu bất thường thì nằm xa hơn các điểm bình thường này [50]. Thuật toán phát hiện bất thường dựa trên khoảng cách phụ thuộc vào hai tham số chính là định nghĩa quan hệ láng giếng dựa trên cái gì và số lượng K láng giềng gần nhất là bao nhiêu. Việc định nghĩa quan hệ láng giềng thường sử dụng hai loại hàm nhân là theo mật độ cục bộ (local density-based kernel mà điển hình là Local Outlier Factor - LOF), và theo khoảng cách (distance-based kernel điển hình là K-Nearest Neighbor - KNN). Do vấn đề tính toán khoảng cách giữa các điểm dữ liệu, do vậy các phương pháp dựa trên khoảng cách đối mặt với các vấn đề về dữ liệu lớn, dữ liệu nhiều chiều. Ngoài ra việc đưa ra giá trị K phù hợp vẫn là một vấn đề cần được làm rõ.

Ghoting và cộng sự [44] đề xuất phương án để tăng tốc độ cho phương pháp dựa trên khoảng cách trong phát hiện bất thường. Phương pháp của họ đạt được hiệu quả ấn tượng, theo đó tốc độ tính toán tuyến tính logarit với kích thước dữ liệu huấn luyện và tuyến tính với số chiều của dữ liệu huấn luyện. Zhang and Wang [44] giới thiệu một phương pháp hiệu quả thông qua sử dụng tìm kiếm động K láng giềng gần nhất trong từng vùng không gian cho dữ liệu nhiều chiều.

Tên là High-dimensional Outlying Subspace Detection (HighDOD). Theo đó, từ điểm dữ liệu đang xét và sử dụng đơn vị đo là trọng số như [8], trọng số này được tính dựa trên tìm kiếm động không gian con chứ không phải là trên một không gian con cụ thể đã được định trước. Việc tìm kiếm động không gian này được xác định dựa vào không gian con, nơi mà điểm đang xét được xem là bất thường (outlier). Họ khẳng định phương pháp hoạt động tốt với dữ liệu khoảng 8 đến 160 chiều, so sánh được với các phương pháp phát hiện bất thường khác.

Local Outlier Factor (LOF) [16] là một đại diện điển hình cho các thuật toán NAD theo hướng dựa trên khoảng cách để phân lớp. Mặc dù một số nghiên cứu phân LOF vào nhóm dựa trên mật độ vì LOF sử dụng độ đo mật độ cục bộ (local density), tuy nhiên bản chất LOF hoạt động dựa trên tính khoảng cách K láng giềng gần nhất. Thuật toán hoạt động theo các bước:

1. Xem xét tập dữ liệu huấn luyện $X = \{x_1, x_2, ..., x_n\} \in \mathrm{R}^d$. Cho mỗi điểm dữ liệu $p \in X$, dist(p,q) là khoảng cách từ p đến một điểm $q \in X$, tham số $D_k(p)$ chỉ khoảng cách lớn nhất từ p đến k láng giềng gần nhất, và $L_k(p)$ thể hiện tập k điểm láng giềng của p

2. Độ vươn, hay khoảng cách có thể tiếp cận được của mỗi điểm dữ liệu p với $q \in L_k(p)$ được tính theo Công thức sau,

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

3. Độ vươn trung bình của mỗi điểm dữ liệu p, ký hiệu $AR_k(p)$,

$$AR_k(p) = MEAN_{o \in L_k(p)} R_k(p, o) \tag{1.2}$$

4. Độ đo LOF cho mỗi điểm p được tính theo Công thức dưới đây,

$$LOF_k(p) = \frac{AR_k(p)}{MEAN_{o \in L_k(p)} AR_k(o)} \tag{1.3}$$

Theo đó, độ đo bất thường cục bộ (LOF) của một điểm dữ liệu p liên quan đến k láng giềng gần nhất là tỉ suất của độ vươn của chính điểm đó với độ vươn

trung bình của các điểm láng giềng ($AR_k(o)$). Các điểm dữ liệu có độ đo LOF cao hơn so với hầu hết các điểm khác trong vùng xem xét có thể được xem là bất thường. Nói cách khác, nếu điểm dữ liệu p bất thường hay dị biệt so với các láng giềng thì cho giá trị LOF càng lớn. Khi sử dụng một ngưỡng quyết định trên vùng giá trị độ đo LOF cho tập dữ liệu đang quan sát, với các điểm lớn hơn ngưỡng sẽ được xem là bất thường, ngược lại được xem là bình thường.

Trong LOF, việc xác định độ vươn $R_k(p,q) = max(dist(p,q), D_k(p))$ dẫn đến chi phí tính toán lớn. Mặc dù thuật toán được khẳng định hoạt động rất hiệu quả với phát hiện bất thường mạng [16], LOF được nhiều nhà nghiên cứu ứng dụng để kết hợp với phương pháp của họ, qua đó tạo ra phương pháp NAD hiệu quả hơn [17],[20]. Tuy vậy, thuật toán vẫn bị xem là hoạt động không ổn định với dữ liệu rất nhiều chiều và phân mảnh (sparity) lớn [106], thêm vào đó, thuật toán vẫn phải cần sự tham gia của chuyên gia trong xác định ngưỡng quyết định.

1.2.1.2. Phương pháp OCC dựa trên mật độ

Các phương pháp phát hiện bất thường dựa trên mật độ (density-based) sử dụng hàm mật độ xác suất với giả định rằng, phân phối đúng của dữ liệu bình thường có thể được sử dụng để đánh giá tính bình thường của dữ liệu. Theo đó, bằng việc đưa thêm tham số ngưỡng cho hàm mật độ xác suất, một điểm đầu vào cho kết quả trên ngưỡng đề ra được xem là dữ liệu bất thường. Trong phạm vi phát hiện bất thường, chỉ dữ liệu bình thường được sử dụng cho huấn luyện. Tuy vậy, vấn đề chính của phương pháp dựa trên mật độ là ước lượng mật độ xác suất của dữ liệu bình thường. Có hai phương pháp ước lượng mật độ xác suất thường hay được sử dụng là Gauss Mixture Models (GMMs) và Kernel Density Estimation (KDE).

Phương pháp GMMs [84] hoạt động dựa trên giả định rằng, dữ liệu bình thường được tạo ra từ tổ hợp của các phân bố Gauss thành phần. Theo đó, GMMs ước lượng hàm mật độ xác suất của dữ liệu bình thường thông qua một số hàm nhân, số lượng các hàm nhân này bé hơn số mẫu dữ liệu huấn

luyện. Trong lĩnh vực NAD, mô hình GMMs được huấn luyện bởi chỉ dữ liệu bình thường. Quá trình kiểm thử, với các điểm dữ liệu có mật độ xác suất dưới ngưỡng đặt ra được xem như là dữ liệu bất thường. Hạn chế của GMMs là yêu cầu lượng lớn dữ liệu huấn luyện để có thể ước lượng tham số cho mô hình.

MP Wand và cộng sự [111] đề xuất giải pháp ước lượng mật độ xác suất có tên là Kernel Density Estimation (KDE), KDE là phương pháp phân lớp dựa trên mật độ. Trong miền ứng dụng OCC, KDE được đánh giá là hiệu quả khi áp dụng cho các tập dữ liệu có mật độ cao, là một trong những thuật toán phổ biến nhất theo hướng dựa trên mật độ. Phương pháp này hoạt động dựa theo hàm ước lượng mật độ xác suất của dữ liệu huấn luyện. Như đã đề cập trước, thuật toán hoạt động không cần bất cứ giả định nào về phân bố xác suất của dữ liệu. KDE ước lượng phân bổ xác suất chưa biết trước của dữ liệu đầu vào, dựa trên dữ liệu huấn luyện bình thường bằng việc sử dụng một số lượng lớn các hàm nhân, thường theo từng điểm dữ liệu.

Cho tập dữ liệu $X = \{x_1, x_2, ..., x_n\} \in \mathrm{R}^d$ nhận được từ một phân bổ xác suất chưa biết trước với hàm mật độ xác suất $p(x)$. Một ước lượng $\hat{p}(x)$ của hàm mật độ xác suất tại mẫu dữ liệu x có thể được tính toán theo Công thức:

$$\hat{p}\,(x) = \frac{1}{n} \sum_{i=1}^{n} K_h\,(x - x_i) \tag{1.4}$$

trong đó $K_h : x \to R$ là hàm nhân với một tham số điều chỉnh h gọi là băng thông (hay độ rộng). Hiệu quả của KDE phụ thuộc trong hai yếu tố là hàm nhân, k, độ rộng h. Có nhiều loại hàm nhân với các đặc điểm khác nhau cho KDE ví dụ như Gauss, Uniform, Exponential. Trong số đó, hàm nhân Gauss (như trình bày bởi Công thức 1.5) là phổ biến, đó là lý do trong phạm vi luận án sử dụng hàm nhân này cho các mô hình KDE. Mỗi điểm dữ liệu đều góp phần vào quá trình ước lượng mật độ trong phương pháp KDE; tham số h điều khiển sự cân bằng giữa độ lệch và phương sai. Giá trị h lớn dẫn đến đường cong phân bố xác suất mịn và ngược lại.

$$K_h(x) = exp(-\frac{x^2}{2h^2}) \tag{1.5}$$

Gần đây, nhiều nghiên cứu sử dụng thuật toán KDE để tạo ra mô hình NAD hiệu quả. Khi ứng dụng KDE theo bài toán OCC, KDE được huấn luyện bởi dữ liệu bình thường; quá trình kiểm thử, nếu điểm dữ liệu cho giá trị mật độ xác suất thấp hơn một ngưỡng định trước thì được xem như là bất thường. Tuy vậy, cũng giống như các phương pháp OCC truyền thống khác, việc xác định ngưỡng quyết định là một vấn đề không hề đơn giản, điều này là yêu cầu tiên quyết đối với mô hình NAD khi được triển khai cho ứng dụng thực tế [18], [40]. Mặc dù KDE được đánh giá là một trong những thuật toán hiệu quả nhất cho phát hiện bất thường, các kết quả nghiên cứu trên cũng cho thấy KDE hoạt động không thực sự ổn định đối với các bộ dữ liệu có độ phức tạp cao, rất nhiều chiều [17].

1.2.1.3. Phương pháp OCC dựa trên vector hỗ trợ

Một trong số phương pháp tiêu biểu cho phương pháp OCC dựa trên vector hỗ trợ là One-class Support Vector Machine (OCSVM) [89], [98]. Mục đích của bài toán OCSVM là tìm ra vector hỗ trợ thể hiện được vùng bao cho các điểm dữ liệu dương (thuộc lớp dữ liệu được huấn luyện).

Với tập dữ liệu $X = \{x_1, x_2, ..., x_n\} \in \mathbb{R}^d$, OCSVM thực hiện ánh xạ X sang một không gian đặc trưng F_k với số chiều lớn hơn bởi một hàm nhân. OCSVM theo hướng siêu phẳng [89] hoạt động dựa trên mục tiêu tìm kiếm một siêu phẳng bao toàn bộ các điểm dương, thường là dữ liệu bình thường về một phía so với trục tọa độ. Với mong muốn, các điểm dữ liệu bất thường trong không gian đặc trưng sẽ nằm ở phần còn lại của siêu phẳng, sát với gốc tọa độ hơn. Tiếp đó, tìm kiếm khoảng cách siêu phẳng (margin) lớn nhất để chia đôi giữa hai

vùng. Hàm mục tiêu cho huấn luyện mô hình có được thể hiện bởi biểu thức 1.6.

$$\min_{w,\ b,\ \xi_i} \frac{1}{2} \|w\|_{F_k}^2 - \rho + \frac{1}{\nu n} \sum_{i=1}^{n} \xi_i \tag{1.6}$$

$$s.t.\ (\omega.\Phi(x_i))) \geq \rho - \xi_i, \xi_i \geq 0, \forall i = 1, ..., n \tag{1.7}$$

trong đó x_i là mẫu dữ liệu thứ i trong tập huấn luyện, n là số mẫu dữ liệu, và Φ là hàm nhân; ω và ρ là tham số vector trọng số và phần bù được đưa vào cho ước lượng trong không gia đặc trưng. Tham số $\xi = (\xi_1, \xi_2, ..., \xi_n)$ cho phép tạo magin có tính linh hoạt hơn. Hàm nhân Gauss thường được sử dụng cho OCSVM. Quá trình tối ưu có thể hiểu là để tăng tối đa số điểm dữ liệu huấn luyện được trả về dương trong không gian đặc trưng, đồng thời tăng kích thước magin. Hiệu quả của OCSVM thường phụ thuộc vào hàm nhân và tham số ν, với tham số $\nu \in (0, 1)$. Nếu ν thấp thì khoảng cách siêu phẳng bé, số điểm trên margin bé và nhiều điểm bình thường có thể rơi vào vùng âm trong không gian đặc trưng; còn ν lớn thì số điểm trên margin lớn, tạo nguy cơ phân lớp sai.

Khi ν tiến đến 1, toàn bộ các điểm dương sử dụng cho huấn luyện đều là vector hỗ trợ; khi đó nếu thuật toán sử dụng hàm nhân Gauss và tham số độ rộng γ mặc định như [98] thì cho kết quả tương tự như phương pháp KDE đề xuất bởi MP Wand và cộng sự [111]. Tuy vậy, OCSVM thường yêu cầu một số lượng lớn dữ liệu bình thường và một số điểm dữ liệu bất thường trong quá trình huấn luyện để có thể nâng cao độ chính xác phân lớp. Thêm vào đó việc ước lượng tham số ν và các tham số khác cho hàm nhân (như tham số độ rộng γ) vẫn là một câu hỏi bỏ ngõ [98]. So với kỹ thuật phát hiện bất thường dựa trên mật độ, phương pháp này có kết quả tương đồng, nhưng khi làm việc với các bộ dữ liệu có kích thước và mật độ lớn, các kỹ thuật dựa trên mật độ như KDE được đánh giá tốt hơn [19].

Tax và Duin [98] đề xuất ra OCSVM theo hướng siêu cầu, với tên gọi Support Vector Data Description(SVDD). Quá trình huấn luyện, dữ liệu huấn luyện (chỉ 1 lớp và gọi là lớp dương) được ánh xạ vào không gian đặc trưng, sau đó tìm

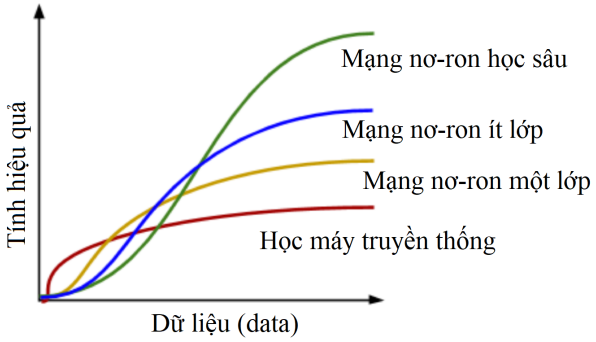
kiếm siêu cầu có bán kính bé nhất chứa tối đa dữ liệu lớp dương. Quá trình kiểm thử, tất cả các dữ liệu đầu vào cho vector đặc trưng nằm ngoài siêu cầu được xem như là dữ liệu bất thường. Họ đề xuất sử dụng tham số ε để loại bỏ bớt các điểm không bình thường của tập dữ liệu huấn luyện, mục đích là để giảm thiểu kích thước siêu cầu. Một số các hàm nhân khác nhau được nhóm tác giả đề xuất, họ khẳng định rằng khi SVDD sử dụng hàm nhân Gauss thì cho kết quả tốt trên nhiều bộ dữ liệu, lúc này phương pháp so sánh được với đề xuất của Schölkopf và cộng sự [89]. Một số phát triển của SVDD với mục đích giúp thuật toán làm việc tốt với cá dữ liệu lớn, nhiều chiều và dạng dữ liệu dòng (streaming data).

1.2.1.4. Phương pháp Centroid

Centroid (CEN) [17] có thể được xem là phương pháp đơn giản nhất trong phát hiện bất thường. Phương pháp này sử dụng hàm nhân Gauss để tạo mô hình NAD từ dữ liệu huấn luyện như sau. Cho tập $X = \{x_1, x_2, ..., x_n\} \in \mathbb{R}^d$ là dữ liệu huấn luyện, μ_j and σ_j là giá trị trung bình và độ lệch chuẩn của thuộc tính thứ j, n là số điểm dữ liệu. Tập X sau đó được chuẩn hoá (normalized) bởi chỉ số z, chỉ số này được tính theo Công thức 1.8.

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

trong đó x_{ij} là giá trị thuộc tính thứ j của điểm dữ liệu x_i, và z_{ij} là chỉ số z tương ứng của nó. Khi kiểm thử, khoảng cách theo Euclid từ điểm dữ liệu đến điểm dữ liệu trung tâm được tính và xem như là độ đo bất thường của dữ liệu kiểm thử. Các kết quả nghiên cứu lai ghép CEN theo sâu bởi các phương pháp phát hiện bất thường khác đã cho thấy hiệu quả ấn tượng [20], ngoài ra CEN là một trong số ít các phương pháp phát hiện bất thường không cần tham số. Tuy vậy, vấn đề sử dụng CEN như là phương pháp độc lập cho phát hiện bất thường không được nhiều nghiên cứu lựa chọn.



Hình 1.4: *Mạng nơ-ron học sâu và các phương pháp truyền thống, Hình từ Alejandro [2016] [6]*

1.2.2. Phương pháp OCC học sâu

1.2.2.1. Học sâu

Học sâu là một nhánh nghiên cứu của học máy, thuật ngữ được nhiều học giả quan tâm trong những năm gần đây, với nhiều định nghĩa khác nhau như tại các nghiên cứu [21], [37], [66], [86]. Nhìn chung, học sâu (Deep learning) là thuật ngữ liên quan đến việc học cách biểu diễn dữ liệu (representation learning) sử dụng một mô hình với nhiều lớp, tầng xử lý [86]. Việc sử dụng nhiều lớp, tầng xử lý dữ liệu giúp cho phương pháp học sâu có thể biểu diễn các dữ liệu rất nhiều chiều (high-demensional data) một cách hiệu quả hơn nhờ khả năng tự học đặc trưng của dữ liệu [37], [86].

Hình 1.4 minh hoạ kết quả khảo sát gần đây [86] về hiệu quả của các phương pháp học sâu so với các phương pháp khác khi kích thước dữ liệu tăng. Theo đó, với dữ liệu ít, phương pháp học sâu không thể hiện rõ hiệu quả, tuy nhiên với dữ liệu tăng cao, hiệu quả của phương pháp học sâu cho khả năng vượt trội so với các phương pháp truyền thống. Điển hình trong số đó như mô hình học sâu sử dụng mạng nơ-ron tích chập (Convolutional Neural Networks-CNN) [60], CNN cho khả năng trích rút được các thuộc tính đặc trưng ẩn trong các câu trúc dữ liệu phức tạp và rất nhiều lớp, và được cho là phù hợp cho cả kiểu dữ liệu tuần tự cũng như dữ liệu hình ảnh. Mạng nơ-ron học sâu GAN (Generative

Adversarial Networks) [48] là một hệ thống gồm hai mạng nơ-ron "cạnh tranh" và tự hoàn thiện nhau, GAN đã tạo ra những yếu tố mới trong lĩnh vực học sâu, sự xuất hiện của GAN đã góp phần tạo ra các mô hinh giả lập tranh, ảnh như thật sử dụng trí tuệ nhân tạo, được biết đến với thuật ngữ "deepfake".

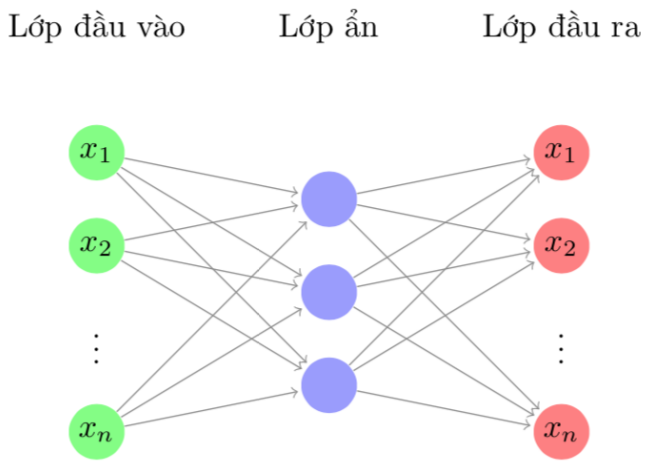
Các mô hình học sâu có thể được phân làm ba nhóm chính: (1) mô hình sinh (unsupervised hay generative learning model), (2) mô hình phân biệt (supervised hay discriminative learning model), (3) mô hình kết hợp (hybrid learning model) [21], [63]. Các mô hình OCC học sâu (Deep - OCC) thuộc nhánh nghiên cứu mô hình sinh học sâu (nhánh 1), một số mô hình phổ biến OCC học sâu (Hình 1.3) như mạng niềm tin sâu (Deep Belief Network - DBN), mạng nơ-ron hồi quy (Recurrent Neural Network - RNN), và AutoEncoder. Trong số đó, học sâu sử dụng kiến trúc AutoEncoder được nhiều các nghiên cứu gần đây ứng dụng cho lĩnh vực an ninh mạng [18], [19], được cho là phương pháp tiên tiến về phát hiện bất thường mạng [20].

AutoEncoder (AE) là một mạng nơ-ron nhân tạo (Artificial Neural Network -ANN) phổ biến và dễ sử dụng [20], [55], là một kiến trúc mạng nơ-ron truyền thẳng, được huấn luyện để tái tạo dữ liệu tại lớp đầu ra như lớp đầu vào.

1.2.2.2. Kiến trúc mạng nơ-ron AutoEncoder

AutoEncoder (AE) có cấu trúc gồm [15], [53] hai khối: mã hoá (lớp đầu vào) và giải mã (lớp đầu ra) được minh hoạ như trên Hình 1.5. Khối mã hoá ánh xạ dữ liệu đầu vào sang không gian lớp ẩn trung tâm (hay còn gọi là tầng cổ chai bottleneck hay vector lớp ẩn). Giả sử f_θ là hàm mã hoá, và $X = \{x_1, x_2, ..., x_n\}$ là tập dữ liệu. Quá trình mã hoá, f_θ sẽ tạo các ánh xạ $x_i \subseteq X$ sang không gian lớp ẩn trung tâm $z_i = f_\theta(x_i)$. Quá trình giải mã, g_θ học để tái tạo dữ liệu đầu ra giống như đầu vào X, $\hat{x}_i = g\theta(z_i)$ từ vector z_i.

Quá trình mã hoá và giải mã thường được trình bày ở dạng hàm số sau: $f\theta(x) = s_f(Wx + b)$ và $g\theta(z) = s_g(W'z + b')$, trong đó $W, W^{'}$ là các ma trận trọng số, b và $b^{'}$ là các ma trận độ lệch, còn s_f và s_g là các hàm kích hoạt tương



Hình 1.5: *Minh họa kiến trúc mạng nơ-ron AutoEncoder*

ứng với quá trình mã hoá và giải mã. Huấn luyện AE là quá trình tối ưu bộ tham số để giảm thiểu lỗi tái tạo (Reconstruction Error -RE) giữa đầu vào x_i và đầu ra tương ứng $\hat{x}_i$. RE có thể được tính toán dựa theo công thức sai số toàn phương trung bình (Mean Square Error- MSE) cho dữ liệu số thực hoặc Entropy chéo (Cross-Entropy) cho dữ liệu nhị phân. Khi sử dụng MSE, giá trị tập hợp các lỗi tái tạo RE có thể được tính như biểu thứ 1.9, biểu thức này thường được xem như là hàm mất mát hay hàm mục tiêu (loss function hay cost function) cho mô hình học máy dựa trên AE.

$$Loss_{AE}(\theta) = \frac{1}{m}\sum_{i=1}^{m}(x_i - \hat{x}_i)^2 \tag{1.9}$$

trong đó θ là tập tham số cho AE, m là số mẫu dữ liệu cho huấn luyện.

Khái niệm hàm mất mát hay hàm mục tiêu trong lĩnh vực học máy, học sâu là hàm số toán học để biểu diễn mức độ khác nhau giữa kết quả dự đoán và giá trị thực của một mô hình. Việc tìm điểm cực tiểu phù hợp cho hàm mất mát có thể được xem là quá trình huấn luyện mô hình học máy.

1.2.2.3. Một số nghiên cứu liên quan AutoEncoder

Có hai hướng ứng dụng kiến trúc mạng nơ-ron AE như sau: (1) Mô hình đơn AE (stand-alone), (2) Mô hình kết hợp AE (hybrid) [20], [37], [107].

Khi sử dụng cho phát hiện bất thường, mô hình đơn AE thường sử dụng RE làm độ đo bất thường. Theo đó, AE được huấn luyện chỉ bởi dữ liệu bình thường để tối thiểu RE $\|x - \hat{x}\|^2$. Khi kiểm thử, với dữ liệu đầu vào bình thường sẽ được mô hình AE cho RE bé, còn dữ liệu bất thường sẽ cho giá trị RE lớn. Hawkins và cộng sự [52] đề xuất mô hình sử dụng AE với ba lớp ẩn bé hơn, huấn luyện mô hình sử dụng chỉ mỗi dữ liệu bình thường. Giá trị RE của mô hình được sử dụng như là chỉ số bất thường, với các điểm dữ liệu đầu vào cho chỉ số RE lớn hơn ngưỡng định trước được xem là bất thường. Mô hình đề xuất được kiểm thử trên bộ dữ liệu Wisconsin Breast Cancer (WBC) và the KDD'99 và được cho là đạt độ chính xác cao. Sakurada và Yairi [87] nghiên cứu ứng dụng AE cho bài toán phát hiện bất thường, cụ thể sử dụng AE như là kỹ thuật để giảm số thuộc tính một cách phi tuyến. Họ so sánh AE truyền thống, DAE (Denoising AE) với các phương pháp phân tích thành phần chính tuyến tính (linear PCA) và phân tích thành phần chính hạt nhân (kernel PCA). Kết quả thử nghiệm trên các bộ dữ liệu nhân tạo và dữ liệu liệu đo đạc không gian, họ khẳng định kết mô hình DAE tốt hơn mô cả linear PCA và kernel PCA về độ chính xác, ngoài ra còn tốt hơn kernel PCA về độ phức tạp tính toán. Fiore và cộng sự [41] xây dựng một kiến trúc của AE gọi là Discriminative Restricted Boltzmann Machines (DRBM), họ giả định rằng dữ liệu bình thường có thể tương tự nhau ở một góc độ nào đó. Họ huấn luyện mô hình đề xuất bởi dữ liệu bình thường, với mong muốn mô hình sẽ hể hiện được tất cả các đặc điểm chung của lưu lượng mạng bình thường. Theo đó, hi vọng mô hình đề xuất sẽ phân biệt được các lưu lượng mạng chưa từng nhìn thấy. Kết quả thử nghiệm trên bộ dữ liệu KDD'99 thể hiện rằng, mô hình của họ có thể làm việc hiệu quả nếu dữ liệu huấn luyện và kiểm thử được thu thập từ cùng một mạng.

Trong mô hình kết hợp sử dụng AE, tầng ẩn trung tâm của một AE có mục đích trong việc nén dữ liệu cũng như biểu diễn đặc trưng dữ liệu [18]. Trong thực tế, nhiều thuật toán phát hiện bất thường gặp thách thức lớn với dữ liệu rất nhiều chiều, dữ liệu phân mảnh, vấn đề này được biết đến với thuật ngữ "curse of dimensionality" [20]. Do vậy, tầng thắt cổ chai AE đã được sử dụng để giảm chiều cho dữ liệu gốc. Nói cách khác, dữ liệu đầu vào được ánh xạ sang không gian có ít chiều hơn, đó là không gian của vector lớp ẩn trong AE. Rajashekar và cộng sự [83] đề xuất kết hợp giữa AE và bản đồ tự tổ chức (Self-Organizing Map - SOM) để mô hình hoá hoạt động bình thường của người dùng điện thoại thông minh. Trong mô hinh này, tác giả sử dụng đầu ra của bộ mã hoá AE để giảm số chiều dữ liệu, sau đó sử dụng SOM cho dữ liệu vector lớp ẩn này để tách thành các cụm người dùng đầu cuối. Nicolau và cộng sự [18] đề xuất phương pháp OCC mới theo hướng mật độ hoá vector lớp ẩn của AE. Mô hình AE được huấn luyện bởi chỉ dữ liệu bình thường, tiếp đó dữ liệu bình thường tiếp tục được kiểm thử bởi mô hình đã huấn luyện để thu thập giá trị mật độ xác suất, giá trị này dựa trên một hàm mật độ xác suất Gauss hay KDE. Trong quá trình kiểm thử, các điểm dữ liệu đầu vào sau khi được AE mã hoá (sinh ra vector lớp ẩn) sẽ được đưa vào mô hình mật độ để phân biệt là bình thường hay không. Veeramachaneni và cộng sự [107] giới thiệu mô hình kết hợp gồm ba phương pháp: mạng nơ-ron AE, mật độ (density-based) và ma trận phân ly (matrix decomposition-based) để tạo thành mô hình phát hiện bất thường. Erfani và cộng sự [37] sử dụng một kiến trúc của AE gọi là mạng niềm tin sâu (Deep Belief Network - DBN) để nâng cao hiệu năng của kỹ thuật phát hiện bất thường khi giải quyết vấn đề dữ liệu rất nhiều chiều "curse of dimensionality". OCSVM sau đó được huấn luyện và nối tiếp phía sau DBN, từ kết quả thử nghiệm, tác giả khẳng định mô hình đề xuất có khả năng giảm số thuộc tính một cách phi tuyến và cho độ chính xác cao hơn OCSVM.

Thời gian gần đây, Cao và cộng sự [20] đề xuất một mô hình gọi là (Shrink AutoEncoder - SAE) cho phát hiện bất thường như đã được đề cập ở phần mở

đầu. Khi áp dụng SAE cho lĩnh vực phát hiện bất thường mạng, mô hình cho kết quả tốt trên nhiều tập dữ liệu kiểm thử (datasets), được cho là mô hình tiêu biểu trong lĩnh vực NAD.

1.2.2.4. Mô hình Shrink AutoEncoder (SAE)

Với mô hình SAE, một thành phần điều chuẩn (regularizer) được thêm vào hàm mất mát của AE. Mục đích của thành phần này là để điều hướng AE trong việc tạo vector lớp ẩn. Mô hình huấn luyện chỉ với dữ liệu bình thường, các điểm dữ liệu này được thành phần điều chuẩn điều hướng để hội tụ về gốc toạ độ (tâm) trong không gian lớp ẩn trung tâm, hay còn gọi là đầu ra mã hoá của AE. Trong nghiên cứu của Cao và cộng sự [20], SAE được thử nghiệp trên nhiều bộ dữ liệu mới và nổi tiếng trong lĩnh vực NAD, nhóm tác giả khẳng định mô hình NAD được tạo từ SAE cho kết quả khả quan, độ chính xác trong phát hiện tốt hơn. Hàm mất mát AE như 1.9 được viết lại cho SAE như sau,

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

Thành phần đầu tiên trong biểu thức 1.10 là RE, thành phần thứ hai là điều chuẩn để dữ liệu lớp ẩn ở tầng trung tâm hội tụ về tâm trong không gian thuộc tính lớp ẩn. Cụ thể hàm mục tiêu của SAE 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 $\hat{x_i}$ và z_i là giá trị tái tạo và vector lớp ẩn ứng với điểm dữ liệu quan sát x_i; m là số mẫu huấn luyện, α là tham số điều chỉnh mức độ cân bằng giữa hai thành phần của hàm mất mát.

Tuy vậy phương pháp học sâu này vẫn tồn tại những hạn chế như: thứ nhất, do thuật toán cố nén và trình bày lại toàn bộ dữ liệu bình thường vào một cụm đơn duy nhất, do vậy thuật toán không hoạt động tốt khi tập dữ liệu cho huấn luyện tồn tại ở dạng nhiều cụm (cluster); thứ hai, mô hình SAE mặc dù cho khả

năng phát hiện bất thường mạng rất tốt, tuy vậy SAE vẫn có thể gặp khó khăn với một số loại tấn công (bất thường), ví dụ kiểu R2L (Remote to Local) [20, xem Bảng 3]. Đây là các mẫu tấn công khi được phân tách (kiểm tra) bởi SAE thường tạo ra các vector được biểu diễn ở gần gốc toạ độ hơn, do vậy việc phân tách giữa bình thường và bất thường khó hơn.

Theo cơ chế hoạt động của SAE, các tấn công mạng mà SAE gặp khó có thể do mẫu dữ liệu có nhiều điểm giống với mẫu dữ liệu bình thường, vì SAE cố ép để dữ liệu bình thường được biểu diễn ở vùng gần gốc toạ độ trong không gian lớp ẩn, do vậy với dữ liệu tấn công gần giống với dữ liệu bình thường cũng sẽ được biểu diễn gần tương tự. Nguyên nhân có thể dẫn đến phương pháp NAD dựa trên học sâu AutoEncoder này có thể không phân tách tốt giữa mẫu bình thường và bất thường trong trong trường hợp nêu trên.

Như vậy, trong phần này đã trình bày khảo sát các phương pháp OCC phổ biến cho NAD trong thời gian gần đây. Kết quả khảo sát cho thấy rất nhiều các nghiên cứu sử dụng OCC cho phát hiện bất thường mạng. Các phương pháp OCC học sâu được cho là lợi thế và phù hợp trong điều kiện sự tăng nhanh của dữ liệu cả về kích thước lẫn độ phức tạp. SAE là mô hình NAD học sâu tiêu biểu, tuy vậy vẫn không thể tránh khỏi một số hạn chế. Việc nghiên cứu NAD được cho là phải liên tục và đổi mới để có thể đáp ứng tốt hơn theo sự tăng lên của đe doạ an ninh mạng. Do vậy, luận án thực hiện nội dung nghiên cứu mô hình NAD dựa trên học sâu theo hướng khắc phục các hạn chế của mô hình tiêu biểu, được trình bày trong Chương 2.

1.3. Phát hiện bất thường dựa trên tổng hợp, kết hợp

Việc tổng hợp hay kết hợp các bộ phân lớp đơn để tạo ra bộ phân lớp mới đã được nhiều các nghiên cứu thực hiện và cho nhiều thành công. Nhìn chung, có ba hướng nghiên cứu chính cho việc kết hợp các bộ phân lớp đơn [13], [39] bao gồm: (1) tổng hợp theo lai ghép (hybrid); (2) Tổng hợp theo học cộng đồng

(ensemble learning); (3) tổng hợp dữ liệu (data fusion).

1.3.1. Tổng hợp theo lai ghép

Bộ phân lớp lai được hình thành trên cơ sở kết hợp hai thành phần, một phương pháp chính và một phương pháp phụ. Có hai chiến lược chính cho hình thành các bộ phân lớp lai. Thứ nhất, thành phần đầu tiên của phương pháp lai ghép trực tiếp xử lý đối với dữ liệu cần quan sát và cho kết quả trung gian (thường được gán nhãn và có số chiều dữ liệu bé hơn). Thành phần thứ hai sau đó sẽ lấy kết quả trung gian như các đầu vào và tạo ra các kết quả sau cùng [13], [18]. Vì các phương pháp lai ghép loại này sử dụng các ưu điểm về tính năng (để giảm chiều dữ liệu) mà không phải là ưu điểm về hiệu quả trong dự đoán, do vậy không phù hợp với mục tiêu nghiên cứu của luận án.

Thứ hai, lai ghép một phương pháp phát signature-based và một phương pháp anomaly-based. Có ba trường hợp xảy ra khi lai ghép như sau: 1) phương pháp dựa trên bất thường nối tiếp sau bởi phương pháp dựa trên dấu hiệu; 2) phương pháp dựa trên dấu hiệu và phương pháp dựa trên bất thường kết nối song song; 3) phương pháp dựa trên dấu hiệu nối tiếp phía sau bởi phương pháp dựa trên bất thường. Trong đó, phương pháp lai ghép thứ ba (3) được cho là hiệu quả và phù hợp với đặc thù khả năng của từng loại kỹ thuật phát hiện [28], [38], [62]. Theo cách đó, hệ thống có thể dựa vào lợi thế của cả phát hiện theo dấu hiệu và phát hiện dựa trên bất thường, qua đó tạo nên phương pháp phát hiện xâm nhập mạng hiệu quả hơn. Tuy vậy, vấn đề cải tiến khả năng cho phương pháp anomaly-based vẫn là bài toán bỏ ngõ, cần tiếp tục được tìm kiếm lời giải.

1.3.2. Tổng hợp theo học cộng đồng

Tổng hợp theo học cộng đồng là thuật ngữ thường được sử dụng trong học máy để thực hiện kết hợp các phương pháp phân lớp đơn với nhau, giúp tạo một bộ phân lớp mới có khả năng tốt hơn. Có ba chiến lược cho kết hợp [13]:

1) đóng bao (bagging), ý tưởng của các phương pháp này là tiến hành xây dựng một lượng lớn các phương pháp phát hiện (thường là cùng loại) trên những tập mẫu huấn luyện khác nhau từ tập huấn luyện gốc thông qua kỹ thuật lấy mẫu lại (resembling). Các phương pháp đơn sẽ được huấn luyện độc lập và song song với nhau nhưng đầu ra của chúng sẽ là các nhãn và thường sử dụng kỹ thuật lấy trung bình hoặc đa số phiếu (majority voting) để cho kết quả cuối cùng; 2) tăng cường (boosting), bằng cách xây dựng một lượng lớn các phương pháp đơn (thường cùng loại). Mỗi mô hình sau sẽ học cách sửa những lỗi của mô hình trước và tạo thành một chuỗi các mô hình. Kết quả cuối cùng thường là kết quả của mô hình sau cùng hoặc là dựa trên phương pháp đa số phiếu; 3) xếp chồng (stacking), ý tưởng là xây dựng một số mô hình (thường là khác loại) và một mô hình tổng, mô hình tổng này thực hiện kết hợp kết quả (là nhãn) từ các mô hình đơn thông qua việc học. Về lý thuyết, phương pháp này có thể xem là tương tự phương pháp tổng hợp dữ liệu dựa trên quyết định được trình bày ở phần sau. Theo Didaci và cộng sự [33], để kết hợp hiệu quả các phương pháp đơn, các phương pháp học theo cộng đồng được xây dựng trên cơ sở huấn luyện các phương pháp đơn trên các tập dữ liệu khác nhau (như bagging hay boosting) thông qua lấy mẫu (resampling) hoặc huấn luyện trên cùng một tập dữ liệu nhưng với bộ đặc trưng khác nhau. Thêm vào đó, sau khi các phương pháp đơn được huấn luyện, phương pháp đa số phiếu thường được sử dụng cho kết hợp thông qua nhãn (label) đã được gán cho các phương pháp đơn khác nhau [13], [36]. Khi áp dụng cho bài toán phát hiện bất thường mạng, không có đủ cơ sở để xác định nhãn vì chỉ có mỗi dữ liệu bình thường được sử dụng cho huấn luyện mô hình [20], [40].

Do vậy, để đạt mục tiêu của luận án là xây dựng được một phương pháp khung cho phát hiện bất thường từ việc kết hợp các phương pháp đơn OCC. Luận án không đi theo hướng kỹ thuật học theo cộng đồng (bagging và bootsting) mà theo hướng tổng hợp dữ liệu (data fusion), gần giống với phương pháp xếp chồng (stacking) trong học theo cộng đồng, nhưng để tổng hợp các bộ phân lớp đơn

OCC. Một số nghiên cứu liên quan đến phương pháp này được trình bày tại phần tiếp theo.

1.3.3. Tổng hợp dữ liệu

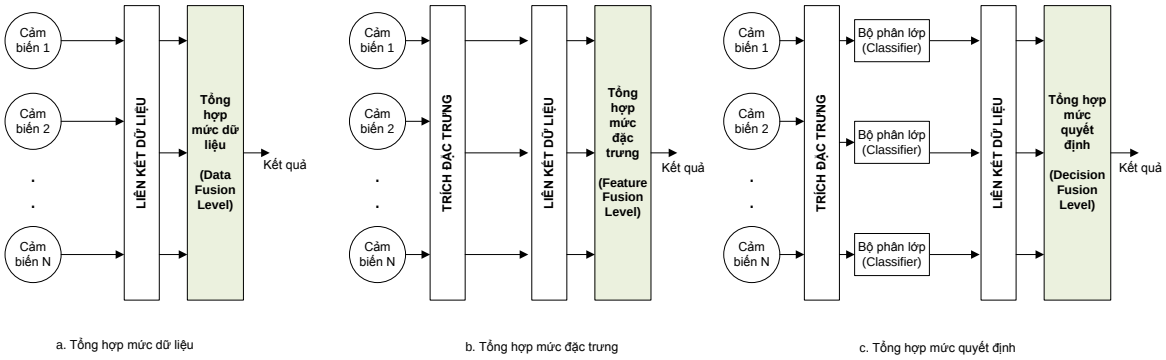
Tổng hợp dữ liệu (Data fusion - DF) được định nghĩa như là một công nghệ cho phép kết hợp thông tin từ nhiều nguồn khác nhau để tạo thành một nguồn duy nhất [10], [59], [103]. Gần đây, công nghệ này đã được áp dụng trong các lĩnh vực khác nhau như phát hiện xâm nhập trái phép, xử lý ảnh hay thiết kế các hệ thống thông minh. DF là một lĩnh vực rộng và có thể được gọi theo nhiều thuật ngữ khác nhau [68], [69], [102]; Định nghĩa của DF cũng có thể được đưa ra bởi các nhà nghiên cứu khi họ áp dụng cho các ứng dụng cụ thể [68]. Trong lĩnh vực phát hiện xâm nhập mạng, DF được định nghĩa là việc xử lý của một nguồn hoặc nhiều nguồn dữ liệu được thu thập từ mạng để cho kết quả đánh giá tốt hơn [102], [117], mục đích cuối cùng của DF khi áp dụng cho lĩnh vực NAD là nâng cao khả năng phát hiện bất thường [69]. Nhìn chung, mô hình tổng hợp (fusion model layer) được thiết kế để làm việc tại một trong ba tầng sau tuỳ vào trường hợp ứng dụng cụ thể: tổng hợp mức dữ liệu (data fusion layer), tổng hợp mức thuộc tính (feature fusion), và tổng hợp mức quyết định (decision fusion layer) [68], [102], [105]. Đầu ra của dữ liệu ở mỗi tầng khác nhau là khác nhau, với DF hoạt động ở tầng dữ liệu hay tầng thuộc tính thường cho giá trị trạng thái, đặc điểm hay tập thuộc tính. Còn đầu ra của DF hoạt động ở tầng quyết định thường là kết quả suy luận (inferences) hoặc quyết định (decision). Các kỹ thuật, phương pháp tổng hợp khác nhau cũng được sử dụng tại các tầng để tăng hiệu quả của bài toán [68]. Hình 1.6 mô tả các mức độ tổng hợp như sau:

- Mức dữ liệu (data fusion layer): được xem là mức tổng hợp thấp nhất, các kỹ thuật (thuật toán) tổng hợp sẽ làm việc với dữ liệu gốc từ các nguồn để tạo nên dữ liệu có nhiều thông tin và độ bao quát hơn. Trong lĩnh vực an ninh mạng, dữ liệu gốc có thể là lưu lượng mạng dạng nhị phân, tệp nhật ký trên các máy tính và thiết bị, các dữ liệu về môi trường (hình ảnh, nhiệt

độ), nguồn điện (điện áp) liên quan đến vùng mạng đang quan sát. Vì làm việc với các nguồn dữ liệu thô dẫn đến độ phức tạp tính toán trong tầng này cao, yêu cầu thêm nhiều kỹ thuật xử lý dữ liệu, đây có thể là lý do rất ít kết quả nghiên cứu cho NAD áp dụng tổng hợp dữ liệu ở mức này [68].

- Mức thuộc tính (feature fusion layer): đây là mức tổng hợp trung gian, mục đích là để giới hạn bộ thuộc tính đặc trưng cho nguồn dữ liệu quan sát. Thường được sử dụng trong giảm số chiều dữ liệu vì vậy thường tăng chi phí tính toán.

- Mức quyết định (decision fusion layer): mức hoạt động này của DF thường được sử dụng cho ra quyết định từ nhiều quyết định cục bộ. Trước khi tổng hợp lại, các bộ phát hiện đơn thường thực hiện các thao tác như tiền xử lý dữ liệu, giảm số chiều và suy luận, ra quyết định cục bộ. Sau đó, các quyết định cục bộ được tổng hợp lại thành quyết định tổng thể (cuối cùng) dựa vào các thuật toán DF. Hoạt động ở mức này giúp cho hệ thống DF có tính linh động hơn nhờ sự độc lập nhất định giữa các quyết định cục bộ và thuật toán tổng hợp. Do đó, chi phí tính toán thường thấp hơn nhiều so với các mức tổng hợp thấp hơn như đã trình bày ở trên. Trong lĩnh vực phát hiện xâm nhập, DF ở mức này được nhiều nhà nghiên cứu quan tâm để tận dụng được sức mạnh của các bộ phát hiện đơn, các kỹ thuật đã hiện hữu [68], [70]. Đó cũng là lý do mà luận án này sử dụng DF ở mức tổng hợp quyết định. Lý thuyết D-S cho phép tính toán trọng số tổng hợp niềm tin cho một dẫn chứng từ nhiều nguồn, đây là cơ sở để phương pháp tổng hợp từ các OCC có thể cung cấp đầu ra ở dạng BL (nhãn), nghĩa là có thể giải quyết khó khăn của vấn đề ngưỡng quyết định mà các phương pháp NAD theo hướng OCC đang gặp phải.



Hình 1.6: *Ba mức tổng hợp dữ liệu, Hình từ [31], [49]*

1.3.4. Tổng hợp dữ liệu dựa trên lý thuyết Dempster-Shafer

1.3.4.1. Lý thuyết Dempster-Shafer

Lý thuyết Dempster-Shafer (D-S) hay còn gọi là lý thuyết dẫn chứng, là một nhánh của toán học, D-S kết hợp các dẫn chứng để tính toán xác suất của một sự kiện. Lý thuyết D-S được giới thiệu từ những năm 1960s bởi Arthur Dempster sau đó được phát triển vào những năm 1970s bởi Glenn Shafer [90].

Lý thuyết này tính toán xác suất của một sự kiện dựa trên kết hợp dẫn chứng thu thập được từ nhiều nguồn khác nhau. Các khẳng định hay phát biểu có thể là một tập con của tập hợp hữu hạn các giả thuyết đặt ra đối với hệ thống đang quan sát. Tập hữu hạn giả thuyết hay tập các nhận định không chắc chắn có thuật ngữ là FoD (Frame of Discernment) và được ký hiệu Θ, đây là tập toàn bộ m trạng thái độc lập có thể xảy ra đối với hệ thống đang xem xét.

$$\Theta = \{H_1, H_2, ...H_m\} \tag{1.12}$$

Cho $E_1, E_2, ..., E_n$ $(n \geq 2)$ là số nguồn dẫn chứng, trong lĩnh vực phát hiện xâm nhập mạng có thể là các bộ phân lớp hay các IDS.

Tập tất cả các trạng thái có thể của Θ được gọi là tập vũ trụ Θ, được định nghĩa là 2^{Θ}. Hàm gán xác suất cơ bản (Basic Probability Assignment - BPA)

qua tập Θ là hàm $m : 2^{\Theta} \rightarrow [0, 1]$ với điều kiện sau:

$$\sum \{m(H))|H \subseteq \Theta\} = 1, m(\emptyset)) = 0 \tag{1.13}$$

trong đó m(H) là trọng số niềm tin được gắn chỉ riêng cho giả thuyết H. Hàm kết hợp DRC (Dempster-Shafer Rule Combination - DRC) của D-S là công cụ để kết hợp trọng số được gắn từ nhiều nguồn quan sát (hay các dẫn chứng).

Khi áp dụng DRC kết hợp hai nguồn E_i và E_j, trọng số niềm tin kết hợp $m(H)$ thu được, $m(H)= (E_i \oplus E_j)(H)$, chi tiết như Công thức sau:

$$m(H) = \frac{\sum_{(B \cap C = H; B, C \subseteq \Theta)} [m_i(B) m_j(C)]}{1 - \sum_{(B \cap C = \varnothing; B, C \subseteq \Theta)} [m_i(B) m_j(C)]} \tag{1.14}$$

Trong trường hợp kết hợp nhiều nguồn thông tin, hàm DRC có biểu thức tổng quát sau:

$$(m_1 \oplus ...m_n)(H) = \frac{\sum_{\cap_i E_i = H} [m_1(E_1) m_2(E_2) ...m_n(E_n)]}{\sum_{\cap_i E_i \neq H} [m_1(E_1) m_2(E_2) ...m_n(E_n)]} \tag{1.15}$$

Tuy vậy, khi thực hiện nhiều nguồn kết hợp, hàm kết hợp DRC cho phép thực hiện kết hợp tuần tự theo từng cặp mà không cần phải thực hiện tổng hợp như tại Công thức 1.15. Giải sử có ba nguồn, khi xem xét BPA cho một giả thuyết $H \subseteq 2^{\Theta}$, giá trị kết hợp $Mass(H)$ từ ba nguồn ,

$$\begin{aligned} Mass(H) &= (m_1 \oplus m_2 \oplus m_3)(H) \\ &= ((m_1 \oplus m_2) \oplus m_3)(H) \\ &= ((m_1 \oplus m_3) \oplus m_2)(H) \\ &= ((m_2 \oplus m_3) \oplus m_1)(H) \end{aligned} \tag{1.16}$$

Trong trường hợp kết hợp niềm tin từ hai nguồn E_i và E_j với trọng số tham gia tương ứng w_i, w_j, hàm kết hợp theo Công thức 1.14 được sửa đổi như sau [69]:

$$(m_i \oplus m_j)(H) = \frac{\sum_{B \cap C = H; B, C \subseteq \Theta)} [w_i m_i(B) * w_j m_j(C)]}{1 - \sum_{B \cap C = \varnothing; B, C \subseteq \Theta)} [w_i m_i(B) * w_j m_j(C)]} \tag{1.17}$$

1.3.4.2. Một số nghiên cứu ứng dụng lý thuyết D-S

Ý tưởng của lý thuyết D-S là đưa ra phương pháp suy luận có lý từ các tri thức luận không chắc chắn. Theo Shafer [90], lý thuyết D-S thu thập mức độ của niềm tin cho mỗi câu hỏi từ vấn đề cần được làm rõ hơn, hoặc là được khẳng định. Sau đó, hàm kết hợp DRC được sử dụng cho kết hợp các mức độ niềm tin này dựa trên các thành phần độc lập của dẫn chứng.

Theo Chen và Aickelin [24], lý thuyết D-S là sự kết hợp giữa lý thuyết niềm tin và xác suất thống kê để tạo ra niềm tin về sự xuất hiện của sự kiện. Họ cho rằng, lý thuyết D-S cập nhật các niềm tin và kết hợp lại để đưa ra một niềm tin duy nhất về trạng thái hay sự kiện tổng thể đang xảy ra. Lý thuyết này được sử dụng để tính toán xác suất của một sự kiện thông qua việc kết hợp dẫn chứng từ nhiều nguồn thông tin. Một phương án hay một phát biểu là một tập con của một tập hữu hạn các giả thuyết đưa ra, tập hữu hạn giả thuyết này có thuật ngữ là Frame of Discernment (FoD) và ký hiệu bởi Θ, đây được xem là tập tất cả các giả thuyết có thể xảy ra của hệ thống. Các khảo sát [26], [34], [68] đã phân tích thuận lợi và khó khăn trong việc sử dụng lý thuyết D-S được các nhà nghiên cứu đưa ra như sau:

- Các điểm thuận lợi: Theo Siaterlis và các cộng sự [93], D-S có những thuật lợi hơn phương pháp suy luận Bayes về độ linh động và tính thực tiễn khi áp dụng, vì Bayes chỉ có thể gán xác suất (niềm tin) cho mỗi trường hợp đơn lẻ trong tập FoD, còn D-S cho phép gán niềm tin cho toàn bộ các trường hợp có thể xảy ra của hệ thống, nghĩa là gán cho toàn bộ các tập con của FoD (2^{Θ}). Thêm vào đó, theo nhóm tác giả, D-S tính toán xác suất của các dẫn chứng hỗ trợ cho một giả thuyết hơn là tính toán xác suất cho chính giả thuyết đó; D-S thuận lợi cho áp dụng với các bài toán với yêu cầu còn lờ mờ và điều kiện chưa xác định. Theo Chen [26], D-S cung cấp phương pháp kết hợp toán học để gom dẫn chứng từ nhiều các quan sát viên khác nhau mà không cần biết trước về xác suất có điều kiện như với suy luận

Bayes. Cũng theo Chen và được phát triển bởi Aickelin [24], D-S rất phù hợp cho bài toán phát hiện bất thường vì không cần phải có trước các tri thức như các phương pháp khác. Họ bổ sung, D-S có thể bỏ qua các dẫn chứng với thông tin cung cấp thực sự không chắc chắn. Theo họ, suy luận Bayes yêu cầu một tri thức tiền định do vậy thường không phù hợp cho bài toán phát hiện bất thường. Đặc biệt là khi bài toán phát hiện bất thường với mục tiêu là khảo sát, phát hiện đối với các trường hợp chưa từng được biết đến. Lúc này, hệ thống không thể dựa vào các tri thức sẵn có.

- Những điểm không thuận lợi: Theo Zadeh [114], Shah và các cộng sự [92], hạn chế lớn nhất của lý thuyết D-S là độ đo niềm tin thì thu thập từ các quan sát các nhau, nhưng luật kết hợp D-S thì lại xem các quan sát này có độ tin cậy như nhau. Do vậy, trong một số ứng dụng sẽ dẫn đến mâu thuẫn trong kết quả. Đặc biệt là nếu có quan sát cho độ tin cậy rất thấp sẽ dẫn đến kết quả của tổng hợp thiếu tin cậy. Theo Siaterlis và cộng sự [93], cùng quan điểm của Chatzigiannakis và cộng sự [23], lý thuyết D-S giả định rằng các các mẫu dẫn chứng là độc lập với nhau như vậy dường như không thực tế. Theo họ, trong thực tế các nguồn thông tin thường có mối liên hệ với nhau nhất định. Theo Chen và Aickelin, độ phức tạp tính toán của D-S sẽ tăng rất nhanh khi số phần tử của FoD lớn. Số hàm BPA sẽ lên đến $2^{|U|}$, với U là tập các phần tử của 2^{Θ} không tính tập rỗng. Số phép tính cho khi thực hiện hàm DRC kết hợp $E_k, E_{k'}$ theo công thức 1.14 lên đến $(2^{|U|-|H|} * 2^{|U|-|H|})$ [58], vì hàm DRC yêu cầu tìm tất cả các cặp $E_k \cap E_{k'} = H$. Trong khi đối với mạng Bayesian độ phức tạp trong trường hợp xấu nhất cũng chỉ $\mathcal{O}(|U|)$ [108]. Thêm vào đó, việc định nghĩa FoD, hàm BPA cho các bài toán ứng dụng phức tạp và khó khăn, điều này là thách thức với nhiều các nhà nghiên cứu [117].

Một số công trình sử dụng D-S cho phát triển các thuật toán trong lĩnh vực phát hiện xâm nhập được mô tả như sau. Tim Bass [10] đề xuất kiến trúc sơ lược cho hệ thống phát hiện xâm nhập theo hướng tổng hợp dữ liệu từ nhiều

nguồn. Trong kiến trúc này, các bước cơ bản cho xây dựng hệ thống theo kiến trúc phân tầng chung được trình bày. Bass nhận định, tổng hợp dữ liệu là tương lai của các hệ thống phát hiện xâm nhập. Giacinto và cộng sự [45] đề xuất mô hình tổng hợp dữ liệu cho phát hiện xâm nhập mạng từ các bộ phân lớp. Trong mô hình đó, bộ thuộc tính đầu vào được tách thành các tập hoàn toàn khác nhau để huấn luyện cho các bộ phân lớp đơn. Quyết định cục bộ từ các phân lớp đơn được kết hợp với nhau bằng một số luật định sẵn hoặc có thể huấn luyện. Sisters và Maglaris [93] trình bày mô hình phát hiện bất thường dựa trên tổng hợp dữ liệu, họ sử dụng lý thuyết D-S như nền tảng cho xây dựng máy phát hiện xâm nhập tấn công DoS. Mô hình được đánh giá trên dữ liệu lưu lượng mạng thật và cho kết quả khá ấn tượng.

Lý thuyết D-S được sử dụng trong nghiên cứu của Hu và cộng sự [54]. Mô hình của họ tập trung giải quyết vấn đề tính toán từ các dữ liệu không chắc chắn. Khi thử nghiệm, họ sử dụng tỉ suất giữa lưu lượng mạng vào và ra với tỉ suất cung cấp dịch vụ như là tiêu chí để phân biệt, thêm vào đó họ sử dụng tri thức có trước từ lĩnh vực DDoS để xây dựng hàm BPA. Theo các nghiên cứu về tổng hợp dữ liệu, Thomas và Balakrishnan [102] nâng cao hiệu năng của hệ thống IDS thông qua sử dụng mô hình tổng hợp dữ liệu từ các IDS. Họ thảo luận vấn đề lựa chọn các nguồn để mô hình DF được hiệu quả hơn. Mô hình sử dụng các IDS theo hướng học có giám sát (suppervised learning) và kiểm thử mô hình trên bộ dữ liệu DARPA 1999. Zhao và cộng sự [117] sử dụng lý thuyết D-S cho kết hợp nhiều phương pháp phát hiện bất thường để tạo ra mô hình NAD dựa trên tổng hợp dữ liệu. Họ đề xuất mô hình theo kiến trúc ba lớp bao gồm: các bộ phát hiện cơ bản (baseline detectors); lớp thông tin (information); lớp tri thức (knowledge). Nghiên cứu của họ tập trung vào lớp thông tin, kết hợp sáu bộ phân lớp được huấn luyện theo phương pháp học có giám sát để tạo mô hình DF. Kết quả thử nghiệm trên bộ dữ liệu KDD-Cup'99 họ khẳng định rằng với các bộ phát hiện có độ tin cậy tương tự nhau sẽ cho kết quả DF tốt hơn. Tuy nhiên, những nghiên cứu của [102], [117] đều được thử nghiệm trên

các bộ dữ liệu cũ, được đánh giá không cao và hiện không còn được khuyến nghị sử dụng cho kiểm thử các công trình về an ninh mạng [97].

Liu và cộng sự [69] đề xuất phương pháp mới cho tối ưu lý thuyết D-S để tổng hợp quyết định từ sáu bộ phát hiện. Phương pháp tổng hợp có tên ODS, dựa trên cải tiến hàm DRC của lý thuyết D-S thông qua bổ sung trọng số cho các phương pháp đơn khi tham gia kết hợp. Các trọng số được xác định dựa trên giả định khoảng cách (Euclid) giữa các điểm dữ liệu bình thường bé hơn khoảng cách giữa điểm dữ liệu bình thường đến điểm dữ liệu bất thường; họ xây dựng hàm BPA của lý thuyết D-S trên giả định đó. Mô hình của họ được kiểm thử trên bộ dữ liệu KDD-Cup'99 và dữ liệu tự xây dựng đã được gắn nhãn. Họ khẳng định, mô hình đề xuất cho độ phát hiện chính xác hơn so với từng phương pháp đơn lẻ.

Chulin Lu và cộng sự [70] đề xuất mô hình NIDS dựa trên tổng hợp dữ liệu theo hướng sử dụng mạng nơ-ron và lý thuyết D-S. Lưu lượng mạng sau khi tiền xử lý được tách theo ba nhóm thuộc tính dữ liệu. Theo đó, ba bộ phân lớp theo hướng học có giám sát được huấn luyện và kiểm thử tương ứng theo ba nhóm dữ liệu này. Mô hình của họ được kiểm thử trên bộ dữ liệu KDD-Cup'99, họ khẳng định tính hiệu quả của mô hình đề xuất. Shah và cộng sự [92] giới thiệu mô hình áp dụng lý thuyết D-S để tổng hợp quyết định có từ bốn bộ phát hiện, gồm hai bộ phát hiện dựa trên bất thường và hai bộ phát hiện dựa trên dấu hiệu. Họ sử dụng bộ dữ liệu KDD-Cup'99 cho kiểm thử và cho rằng mô hình đề xuất có kết quả khả quan. Gần đây, Mattar và cộng sự [73] đề xuất mô hình phát hiện bất thường cho lưu lượng mạng theo hướng ứng dụng lý thuyết D-S. Họ đưa ra phương pháp định nghĩa hàm BPA chính là giá trị Specificity (SP) thu được từ các bộ phân lớp. Họ thực nghiệm kết hợp hợp bốn bộ phân lớp khác nhau để nâng cao khả năng phát hiện. Tuy nhiên, giải pháp được nhóm tác giả đề xuất theo hướng kết hợp các bộ phân lớp được huấn luyện theo học có giám sát.

Trong khảo sát gần đây Li và cộng sự [68] trình bày một số kết quả nghiên cứu sử dụng kỹ thuật tổng hợp dữ liệu từ nhiều nguồn cho xây dựng hệ thống

phát hiện xâm nhập mạng. Các thống kê của họ cho thấy, cơ bản các nghiên cứu sử dụng mô hình DF lớp ra quyết định (decision layer) để giải quyết vấn đề đặt ra. Thêm vào đó, họ gợi ý về việc nghiên cứu cá mô hình DF cho giải quyết bài toán phát hiện bất thường mạng, các nghiên cứu nên được kiểm thử trên các bộ dữ liệu có tính mới và hiện đại như UNSW-NB15.

1.4. Đánh giá giải pháp

Để thực nghiệm đánh giá một giải pháp đề xuất trong lĩnh vực phát hiện bất thường, hai yếu tốt chính cần quan tâm gồm bộ dữ liệu cho kiểm thử và các chỉ số đánh giá được sử dụng. Phần này sẽ trình bày các yếu tố này, các tập dữ liệu và các chỉ số được mô tả đây được sử dụng cho các thực nghiệm của luận án.

1.4.1. Bộ dữ liệu cho kiểm thử

Bộ dữ liệu sử dụng cho kiểm thử các giải pháp an ninh mạng thường chứa các bản ghi, thể hiện thuộc tính đặc trưng của lưu lượng mạng, với các trường dữ liệu đặc trưng theo điều kiện thu thập, mỗi bản ghi ứng với nhãn được gán. Trong phạm vi luận án, các kết quả nghiên cứu lý thuyết cũng đã được đánh giá qua thực nghiệm, các bộ dữ liệu được sử dụng cho thực nghiệm đều phổ biến và nổi tiếng trong lĩnh vực học máy và an ninh mạng, cụ thể gồm:

- Bộ dữ liệu KDD Cup 1999: Đây từng là bộ dữ liệu phổ biến cho kiểm thử các công trình nghiên cứu về lĩnh vực IDS trong hai thập kỷ qua. Dataset KDD Cup 1999 là một phiên bản của bộ dữ liệu DARPA 1998 [97], được sử dụng trong cuộc thi Khai phá dữ liệu và khảo sát tri thức quốc tế lần thứ 3 (The Third International Knowledge Discovery and Data Mining Tools Competition). Để tạo ra bộ dữ liệu này, các thuộc tính từ bộ dữ liệu thô của dataset DARPA được trích ra thành các đặc trưng theo các thuật toán riêng biệt, độ lớn và số thuộc tính của bộ dữ liệu cũ vẫn được giữ nguyên [80]. Hạn chế của KDD Cup 1999 [97] nằm ở hai điểm chính: bộ dữ liệu có

rất nhiều bản ghi trùng lặp, cụ thể trên bộ dữ liệu huấn luyện và kiểm thử tương ứng có 78% và 75% bản ghi trùng; thêm vào đó, sự không đồng đều trong phân bố giữa tập huấn luyện và tập kiểm thử làm ảnh hưởng đến kết quả đánh giá cho các thuật toán phân lớp. Theo các đánh giá [97], khi sử dụng các bộ phân lớp phổ biến J48, Decision Tree Learning, Naive Bayes, NBTree, Random Forest, Support Vector Machine (SVM)... để huấn luyện và kiểm thử trên bộ dữ liệu KDD thì cho độ chính xác rất cao, tất cả đều từ 96-98%, do vậy việc sử dụng bộ dữ liệu này cho kiểm thử các thuật toán mới hơn sẽ không còn thực sự phù hợp nữa.

- NSL KDD: là bộ dữ liệu được Tavallaee và cộng sự công bố năm 2009 [97], là một phiên bản được định nghĩa lại từ bộ KDD Cup 1999, trên cơ sở loại bỏ một số bản ghi bị thừa, trùng lặp thông tin [32]. Hiện tại, bộ dữ liệu được sử dụng trong rất nhiều công trình nghiên cứu về học máy và an ninh mạng. So với bộ dữ liệu gốc, bộ dữ liệu này có các đặc điểm mới như: không bao gồm các bản ghi dư thừa trong tập huấn luyện, do vậy kết quả phân lớp sẽ không theo hướng của các bản ghi xuất hiện nhiều hơn; không còn bản ghi trùng lặp trong bộ dữ liệu kiểm thử; xử lý vấn đề khi vùng kết quả đánh giá hẹp hiệu quả hơn so với bộ dữ liệu KDD; cân đối hợp lý số lượng bản ghi giữa tập huấn luyện và kiểm thử. Bộ dữ liệu này được đánh giá cao trong đánh giá các thuật toán học máy, hạn chế lớn nhất của bộ dữ liệu đó là không thể hiện được vết của các tấn công ở mức độ thấp, tinh vi [35].

- Bộ dữ liệu UNSW-NB15 [75]: được công bố năm 2015, được tạo thông qua việc thu thập lưu lượng mạng được thiết lập bởi phòng thí nghiệm Cyber Range của Australian Centre for Cyber Security (ACCS). Hệ thống mạng và giả lập tấn công được đánh giá là sát với thực tế hoạt động của mạng và các mã độc hiện nay thông qua công cụ giả lập tấn công của hãng IXIA. Sau khi sử dụng Tcpdump để thu thập hơn 100 GB lưu lượng thô (dạng tệp Pcap), với 9 mẫu tấn công (Fuzzers, Analysis, Backdoors, DoS, Exploits,

Generic, Reconnaissance, Shellcode và Worms), nhóm tác giả sử dụng công cụ Argus, Bro-IDS với 12 thuật toán khác nhau để tạo ra 49 thuộc tính dữ liệu [75]. Bộ dữ liệu UNSW-NB15 được nhiều công trình nghiên cứu sử dụng để kiểm thử các thuật toán phân lớp trong những năm gần đây [35], nhờ khắc phục được hạn chế thiếu các mẫu tấn công mới; lưu lượng mạng thể hiện được dịch vụ mạng đương thời; có sự phân bố đồng đều giữa tập huấn luyện và kiểm thử (thường phân bố theo tỷ lệ 40/60 tương ứng giữa tập kiểm thử và tập huấn luyện) [76].

- Các bộ dữ liệu CTU13: được nghiên cứu bởi Đại học Kỹ thuật Séc và được công bố năm 2011 [42]. Đây là các bộ dữ liệu chứa thông tin bao gồm cả lưu lượng các Botnet, dữ liệu bình thường và dữ liệu lưu lượng của hạ tầng dịch vụ mạng. Bộ dữ liệu gồm 13 tập dữ liệu con theo các tình huống hoạt động khác nhau ứng với từng mẫu mã độc. Các gói tin sau khi được thu thập (dạng .pcap) sẽ được xử lý bởi công cụ Argus (Audit Record Generation and Utilization System) để tạo thành các thuộc tính cho bộ dữ liệu huấn luyện và kiểm thử. Các bộ dữ liệu con có số các thuộc tính khác nhau và được đánh tên theo ký hiệu từ CTU13_01 đến CTU13_13. Bộ dữ liệu hiện sẵn có tại trang Website của đơn vị chủ quản, phổ biến được sử dụng cho các thực nghiệm gần đây là các bộ dữ liệu con CTU13_08, CTU13_09, CTU13_10, CTU13_13 [17]. Hạn chế lớn nhất là bộ dữ liệu chỉ chứa các tấn công mạng dạng Botnet. Trong thực nghiệm của luận án, thực hiện chia tách ngẫu nhiên từng bộ dữ liệu con theo kích thước 4/6 của tập gốc và chỉ giữ lại dữ liệu bình thường cho tập huấn luyện, tập kiểm tra còn lại tương ứng với 60% tập dữ liệu gốc.

- Tập dữ liệu BoT-IoT [61] được tạo bởi môi trường mạng thật tại tại phòng thí nghiệm Cyber Range của Trung tâm nghiên cứu an ninh mạng UNSW Canberra Cyber. Tập dữ liệu bao gồm các loại tấn công DDoS, DoS, OS and Service Scan, Key-logging and Data ex-filtration. Trong luận án, để

phù hợp với điều kiện cấu hình thử nghiệm hiện có, luận án sử dụng tập rút gọn từ tập dữ liệu gốc (5%, kích thước gốc là 3 triệu bản ghi). Sau đó, chọn 20% của dữ liệu bình thường cho tập huấn luyện còn tập kiểm tra gồm 80% của bình thường và toàn bộ các bất thường. Ở đây, bất thường tương ứng với toàn bộ các loại tấn công. Các lựa chọn trên đều được thực hiện theo phương pháp ngẫu nhiên.

- Tập dữ liệu Spambase và tập dữ liệu InternetADs. Đây là các tập dữ liệu từ kho dữ liệu cho học máy (UCI Machine Learning Repository [9]). Tập dữ liệu Spambase là tập các thư điện tử rác (spam e-mails), bao gồm 57 thuộc tính, tập huấn luyện bao gồm (2230 bản ghi) và tập kiểm tra (558 bình thường và 363 bất thường). Tập InternetADs là tập các quảng cáo trên mạng Internet, bao gồm 1558 thuộc tính, tập huấn luyện gồm (1582 bản ghi) và tập kiểm tra (396 bình thường và 77 bất thường).

- Tập dữ liệu WUSTL-IIOT-2018 ICS (SCADA) Cybersecurity [99] được tạo cho mục đích nghiên cứu an ninh mạng SCADA và được thiết kế theo điều kiện môi trường mạng thật SCADA. Tập dữ liệu chỉ gồm 6 thuộc tính, bao gồm tập dữ liệu bình thường (505921 bản ghi) và tập bất thường (13750). Luận án tạo tập huấn luyện bằng cách lấy 10% của tập bình thường cho tập huấn luyện; lấy 90% của dữ liệu bình thường còn lại, gộp với toàn bộ dữ liệu bất thường để sinh ra tập kiểm tra.

Trong luận án khi thực nghiệm sử dụng các bộ dữ liệu trình bày trên, kỹ thuật mã hoá nhị phân (bit) hoá dữ liệu (One-hot) được sử dụng cho các trường dữ liệu tập hợp (categorical features). Để thử nghiệm trong điều kiện bài toán OCC, tập huấn luyện và tập kiểm thử đều là dữ liệu bình thường (trường hợp có sử dụng tập kiểm thử, chia tách tập huấn luyện theo tỉ lệ 70/30 tương ứng với dữ liệu cho huấn luyện và dữ liệu cho kiểm thử). Quá trình thực nghiệm toàn bộ nhãn của các tập này đều không được sử dụng. Khi kiểm tra các mô hình, trong các tập dữ liệu kiểm tra, tất cả các loại tấn công mạng của các tập

dữ liệu đều được xem là dữ liệu bất thường và được gán nhãn là 1, còn dữ liệu bình thường được gán nhãn là 0. Toàn bộ nhãn trong tập kiểm tra đều được bỏ đi tại thời điểm kiểm tra, nhãn này chỉ sử dụng sau khi hoàn tất quá trình thực nghiệm, để giúp so sánh, đánh giá kết quả đầu ra các mô hình phát hiện bất thường. Các bộ dữ liệu trên là phổ biến trong lĩnh vực an ninh mạng [13], [17] [35], được sử dụng xuyên suốt luận án cho đánh giá các kết quả nghiên cứu.

1.4.2. Các chỉ số đánh giá

Về cơ bản, có hai nhóm chỉ số đánh giá một kỹ thuật phát hiện bất thường. Đầu tiên là hiệu năng, đây là phương pháp để ước lượng mức độ tài nguyên cần thiết cho thuật toán sử dụng, thường bao gồm CPU và bộ nhớ. Thứ hai là hiệu quả, thường chỉ ra mức độ về khả năng của thuật toán. Đối với bài toàn phát hiện bất thường, các chỉ số hiệu quả để chỉ khả năng phát hiện (prediction ability or detection ability) của hệ thống, nghĩa là khả năng phân biệt giữa các đối tượng bình thường và bất thường. Thêm vào đó, chỉ số độ ổn định của khả năng phát hiện cũng được sử dụng để đánh giá.

Ngày nay, với sự phát triển nhanh của công nghệ phần cứng, cụ thể là các bộ vi xử lý [102], [103], hầu hết các nhà nghiên cứu trong lĩnh vực phát hiện bất thường chỉ tập trung vào nghiên cứu cải tiến, nâng cao khả năng phát hiện của hệ thống. Tuỳ theo loại dữ liệu đầu ra của phương pháp NAD có thể cung cấp (là AS hay BL) mà các chỉ số tương ứng thường được sử dụng.

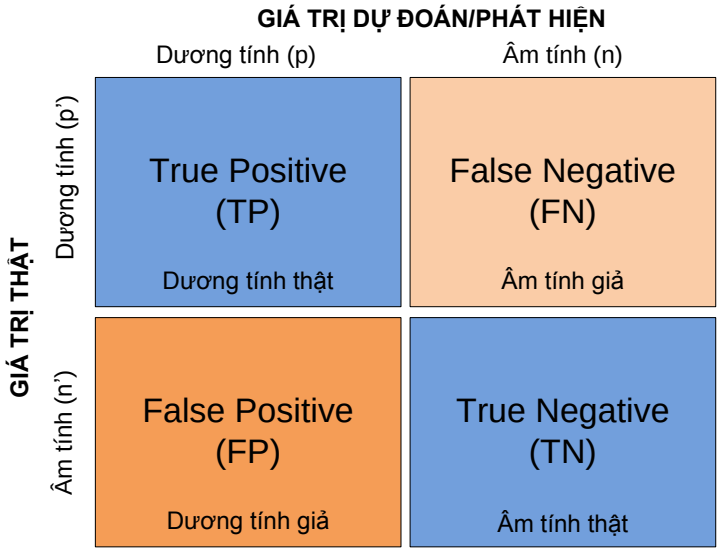
1.4.2.1. Chỉ số đánh giá với đầu ra là nhãn nhị phân

Độ chính xác (Accuracy - ACC): Chỉ số này như là tỉ lệ giữa các dữ liệu được phân loại đúng trên toàn bộ dữ liệu, trong bài toán OCC đó là bao nhiêu mẫu dữ liệu được phân loại đúng là bất thường, là bình thường trên tổng số tất cả cá mẫu dữ liệu [79]. Công thức tính toán ACC như sau,

$$Accuracy(ACC) = \frac{TP + TN}{TP + FP + FN + TN} \tag{1.18}$$

Trong đó, các giá trị TP FP TN và FN được tính toán bởi ma trận lỗi 1.7. Trong bài toán sử dụng ACC như chỉ số so sánh, chỉ số ACC cao ứng với mô hình đó được đánh giá tốt hơn.

Ma trận lỗi (Confusion Matrix): Cách tính sử dụng chỉ số ACC như ở trên chỉ cho biết được bao nhiêu phần trăm lượng dữ liệu được phân loại đúng mà không chỉ ra được cụ thể mỗi loại được phân loại như thế nào. Do vậy, đánh giá một phương pháp OCC có thể sử dụng ma trận lỗi (Confusion matrix) kích thước (2 x 2) như Hình 1.7, trong đó các hàng thể hiện giá trị thật, các cột thể hiện giá trị dự đoán [20], [57]. Phương phát phát hiện bất thường mạng máy tính là để phân biệt giữa lưu lượng mạng đang xét là bất thường hay bình thường. Khi sử dụng các phương pháp OCC cho phát hiện bất thường, lớp bình thường (Normal) có thể xem là lớp âm tính (negative), dữ liệu không thuộc lớp bình thường được xem là bất thường (Anomaly), là lớp dương tính (positive).



Hình 1.7: *Ma trận lỗi (Confusion Matrix).*

Khi hoạt động, hệ thống sẽ đưa ra cảnh báo hoặc không. Các cảnh báo có thể là đúng hay sai, một số thuật ngữ được đưa ra để biểu thị các chỉ số này như sau:

- True positive (TP): Là số các tấn công hay bất thường được hệ thống phát

hiện ra, gọi là dương tích thật.

- False positive (FP): Là số các điểm dữ liệu bình thường nhưng được hệ thống đưa ra cảnh báo, gọi là dương tính giả.
- True Negative (TN): Là số các điểm dữ liệu bình thường và được hệ thống nhận ra và không đưa ra cảnh báo, gọi là âm tính thật.
- False Negative (FN): Là số các tấn công hay bất thường nhưng hệ thống không phát hiện ra, gọi là âm tính giả.

Tỉ lệ phát hiện và tỉ lệ cảnh báo sai: Ngoài ACC, cặp chỉ số cũng thường được sử dụng cho đánh giá độ chính xác của phân lớp là DR và FAR. Tỉ lệ phát hiện (Detection Rate - DR) là tỉ lệ giữa tổng số tấn công/bất thường được phát hiện đúng trên tổng số tấn công [79], DR được tính toán theo Công thức sau,

$$DR = \frac{TP}{TP + FN} \tag{1.19}$$

Tỉ lệ phát hiện sai (False Alarm Rate - FAR) là tỉ lệ giữa số điểm dữ liệu bình thường bị đưa ra cảnh báo trên tổng số điểm dữ liệu bình thường. FAR được tính theo Công thức sau,

$$FAR = \frac{FP}{FP + TN} \tag{1.20}$$

Theo đó, khi xem xét cùng mức FAR, nếu bộ phân lớp nào cho DR tốt hơn thì bộ phân lớp đó được đánh giá hiệu quả hơn.

Độ đo F1-Score: Khi áp dụng cho các bài toán thực tế, đặc biệt là bài toán về phát hiện bất thường, thường có sự chênh lệch lớn giữa số lượng điểm dữ liệu bình thường và bất thường. Hơn thế nữa vấn đề phát hiện sai đối với dữ liệu bất thường được ưu tiên hơn. Do vậy việc sử dụng các đơn vị đo như ACC hay DR, FAR có những hạn chế [79]. F1-score là đơn vị đo để khắc phục các hạn chế đó [82], F1-score được tính dựa trên hai khái niệm khác là: precision và recall.

Trong bài toán OCC, Precision được định nghĩa là tỉ lệ số điểm dương tính thật trong số những điểm được phân loại là dương tính (TP + FP). Recall được

định nghĩa là tỉ lệ số điểm dương tính thật trong số những điểm thực sự là dương tính (TP + FN), theo các Công thức,

$$precision = \frac{TP}{TP + FP} \tag{1.21}$$

$$recall = \frac{TP}{TP + FN} \tag{1.22}$$

Và F1-score được tính theo Công thức,

$$F1 - score = 2\frac{1}{\frac{1}{precision} + \frac{1}{recall}} = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{1.23}$$

F1-score là chỉ số đo cân bằng giữa precision và recall và được xem là chỉ số chính để đánh giá hiệu quả của các mô hình (thuật toán) phát hiện bất thường với đầu ra là nhãn nhị phân [13], [68], [69]. Giá trị F1-score cao thể hiện mô hình cho khả năng phát hiện bất thường tốt hơn.

1.4.2.2. Chỉ số đánh giá với đầu ra là độ đo bất thường

Đường cong ROC và AUC: Khi phương pháp phân lớp không thể đưa ra được nhãn nhị phân mà là một độ đo bất thường, chỉ số thường sử dụng cho đánh giá các phương pháp trong trường hợp này là ROC và AUC. Đường cong ROC (Receiver Operating Characteristic ROC) là đơn vị đo được đề xuất để thể hiện sự cân bằng của DR và FAR [79], [96]. ROC minh hoạ mối quan hệ giữa DR và FAR cho một bộ phân lớp cụ thể. Đường cong ROC có được từ hai tham số này qua rất nhiều các ngưỡng và được tính theo công thức sau [95].

$$ROC = \frac{P(x|positive)}{P(x|negaitive)} \tag{1.24}$$

Đỉnh của đường cong ROC hướng đến giá trị góc (0,1) trên trục toạ độ thể hiện thuật toán tương ứng được đánh giá hiệu quả hơn [13].

AUC (Area Under Curver) là vùng diện tích dưới đường cong ROC, chỉ số này minh hoạ chất lượng phân lớp của một mô hình học máy, chất lượng này được xác định gần như trung bình trên nhiều ngưỡng khác nhau. Một mô hình phân lớp tốt nếu AUC tiến đến sát 1, có nghĩa là mô hình có khả năng phân biệt các lớp dữ liệu đang quan sát rất tốt. AUC được sử dụng phổ biến khi đánh giá các thuật toán phân lớp khác nhau mà ở đó chưa xác định được cụ thể ngưỡng quyết định [20].

1.4.2.3. Độ ổn định

Độ ổn định của mô hình trên các môi trường mạng khác nhau cũng được xem là một trong những chỉ số đánh giá quan trọng đối với một giải pháp phát hiện bất thường mạng. Khi xem xét độ chính xác (ví dụ F1-score, ACC) của một giải thuật trên các đối tượng quan sát (tập dữ liệu) khác nhau, nếu chỉ số được đánh giá có giá trị ổn định hơn các phương pháp phát hiện bất thường khác trên đa số trường hợp thì mô hình phân loại đó được đánh giá là tốt hơn [13].

1.5. Kết luận

Chương này trình bày bốn phần chính, trình bày nội dung kiến thức cơ sở và một số nội dung liên quan của luận án. Trong phần thứ nhất, giới thiệu một số khái niệm liên quan, trình bày mô hình tổng quan NAD; làm rõ lý do phương pháp huấn luyện mô hình NAD theo học bán giám sát là phù hợp, nội dung trong phần cũng trình bày hai loại đầu ra phổ biến của mô hình NAD là "Độ đo bất thường" và "Nhãn nhị phân".

Phần thứ hai trình bày một số phương pháp đơn OCC phổ biến cho NAD như KDE, LOF, OCSVM. Tiếp đó giới thiệu một số kết quả nghiên cứu NAD dựa trên học sâu, tập trung giới thiệu mô hình học sâu tiêu biểu cho NAD, mô

hình SAE. Nội dung trình bày khẳng định phương pháp phát hiện bất thường dựa trên mạng nơ-ron học sâu là tiên tiến hiện nay. Từ kết quả phân tích, nội dung nghiên cứu, phát triển các phương pháp đơn cho phát hiện bất thường dựa trên học sâu sẽ được luận án trình bày tại Chương 2.

Phần thứ ba giới thiệu về các phương pháp kết hợp, tổng hợp từ các phương pháp đơn để tạo mô hình đồng nhất, hiệu quả. Trình bày kết quả khảo sát, phân tích lý do phương pháp tổng hợp dữ liệu (Data Fusion) là phù hợp cho mục tiêu luận án đề ra. Thêm vào đó, phần này đi sâu trình bày lý thuyết D-S và các nghiên cứu liên quan. Khẳng định, lý thuyết Dempster-Shafer (D-S) được đánh giá là phù hợp cho bài toán phát hiện bất thường nhờ sự linh hoạt và không yêu cầu tri thức tiền định khi xây dựng mô hình.

Phần còn lại trình bày về một số yếu tố chính cho thực nghiệm đánh giá thuật toán phát hiện bất thường. Đầu tiên giới thiệu về các bộ dữ liệu phổ biến cho lĩnh vực an ninh mạng, giới thiệu cách thức luận án sử dụng các bộ dữ liệu (10 bộ) cho kiểm thử các thuật toán OCC. Tiếp đó trình bày về các chỉ số đo lường thường được sử dụng cho đánh giá, so sánh các phương pháp phân lớp hay các thuật toán phát hiện bất thường. Các chỉ số đánh giá được phân nhóm theo dạng đầu ra của mô hình NAD, ngoài ra chỉ số cho đánh giá sự ổn định của một mô hình NAD cũng được đề cập. Nội dung trình bày trong phần sẽ được sử dụng tại các Chương 2, 3 của luận án.

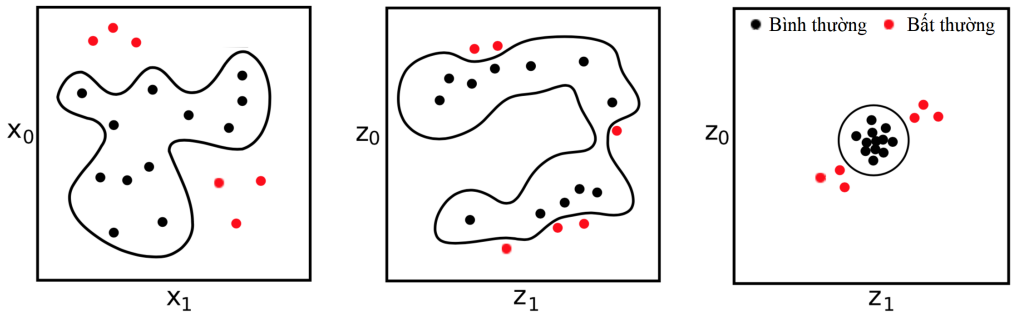
CHƯƠNG 2. PHÁT HIỆN BẤT THƯỜNG DỰA TRÊN HỌC SÂU AUTOENCODER

Chương này trình bày kết quả nghiên cứu phát triển mô hình phát hiện bất thường dựa trên học sâu, nội dung trình bày trong bốn phần. Phần đầu, giới thiệu một số hạn chế mà phương pháp học sâu tiêu biểu NAD có thể đang gặp phải. Tiếp đó, trình bày về phương pháp giải quyết vấn đề đặt ra thông qua cải tiến, phát triển từ mô hình tiêu biểu đang có. Trong phần ba, trình bày về thực nghiệm, kết quả và đánh giá giải pháp đề xuất thông qua các bộ dữ liệu phổ biến cho lĩnh vực an ninh mạng. Trong phần cuối, trình bày kết luận chương. Kết quả nghiên cứu trong chương được công bố trên các công trình [CT1], [CT5] (trong phần CÁC CÔNG TRÌNH CÓ LIÊN QUAN ĐẾN LUẬN ÁN).

2.1. Giới thiệu

Như đã trình bày ở phần mở đầu, mạng nơ-ron học sâu dựa trên kiến trúc AutoEncoder (Deep AutoEncoder - DeAE) được nhiều học giả quan tâm nghiên cứu, DeAE có thể khắc phục các hạn chế của các phương pháp truyền thống và được cho là phương pháp tiên tiến (the-state-of-the-art) cho phát hiện bất thường mạng [52], [55], [87]. AutoEncoder (AE) là một mạng nơ-ron truyền thẳng được huấn luyện để tái tạo đầu ra giống với đầu vào [15], [53]. DeAE hình thành từ việc sử dụng AE với nhiều lớp ẩn, tầng lớp ẩn trung tâm đóng vai trò nén dữ liệu đầu vào sang không gian thuộc tính có số chiều thấp hơn theo hướng, giữ lại thông tin quan trọng và bỏ đi các thông tin thừa từ dữ liệu gốc ban đầu [17]. Các nghiên cứu gần đây về AE sử dụng dữ liệu tầng ẩn trung tâm làm đại diện đặc trưng cho dữ liệu đầu vào (Feature Representation - FtR). Nhờ đó giúp cho mô hình giải quyết vấn đề dữ liệu nhiều chiều [18], [20], [83]. Mô

hình Shrink AE (SAE) [20] được cho là mô hình tiêu biểu trong phát hiện bất thường mạng. Xét về khía cạnh huấn luyện mạng, SAE là một mở rộng của AE truyền thống thông qua sử dụng một tham số điều chuẩn vào hàm mất mát của AE. SAE được huấn luyện để đồng thời thực hiện hai mục tiêu là tái tạo dữ liệu đầu ra từ đầu vào và buộc các dữ liệu FtR hội tụ về gốc toạ độ. Minh họa cho



Hình 2.1: *Minh hoạ phân bố dữ liệu: (a) không gian gốc, (b) không gian vector lớp ẩn AE, (c) không gian vector lớp ẩn của SAE, Hình từ [20].*

bản chất hoạt động của SAE như tại Hình 2.1. Trong đó, Hình 2.1(a) thể hiện không gian dữ liệu đầu vào gốc. Hình 2.1(b) trình bày không gian thuộc tính của lớp ẩn trung tâm của mô hình AE bình thường, và Hình 2.1(c) trình bày không gian thuộc tính của lớp ẩn trung tâm của mô hình SAE. Qua đó thể hiện dữ liệu bình thường được ràng buộc để phân bố trong một vùng không gian nhỏ gần với gốc toạ độ.

Mặc dù DeAE mà cụ thể là SAE đã được chứng minh cho hiệu quả phát hiện bất thường tốt trên nhiều tập dữ liệu kiểm thử phổ biến [20], phương pháp này hiện vẫn có thể gặp những hạn chế nhất định. (i) Việc SAE được huấn luyện để nén tất cả dữ liệu huấn luyện vào một cụm (cluster) đơn trong không gian vector lớp ẩn, do vậy SAE có thể đạt hiệu quả không cao với trường hợp đối tượng quan sát có dữ liệu trạng thái bình thường tồn tại ở dạng nhiều cụm. (ii) Mô hình SAE mặc dù cho khả năng phát hiện bất thường mạng rất tốt, tuy vậy SAE vẫn có thể gặp khó khăn với một số loại tấn công (bất thường). Trong tình huống này, các mẫu tấn công khi được kiểm tra bởi mô hình SAE thường tạo ra

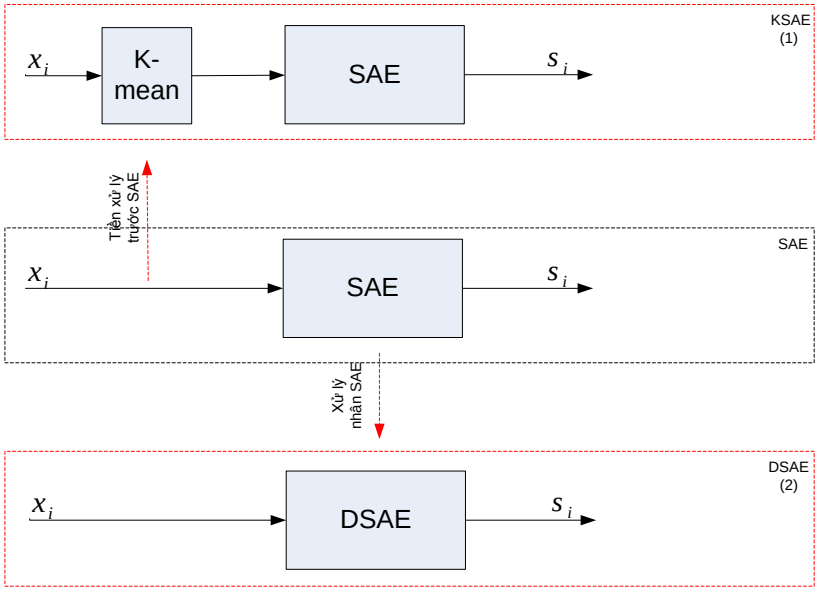
các vector lớp ẩn có xu hướng gần gốc toạ độ hơn, do vậy việc phân tách giữa bình thường và bất thường khó hơn. Các tấn công mạng mà SAE gặp khó có thể do mẫu dữ liệu có nhiều điểm giống với mẫu dữ liệu bình thường, vì SAE cố ép để dữ liệu bình thường được biểu diễn ở vùng gần gốc toạ độ trong không gian lớp ẩn, do vậy với dữ liệu tấn công gần giống với dữ liệu bình thường cũng có thể cho lỗi tái tạo (Reconstruction Errors - RE) bé, và có cách trình diễn dữ liệu trong không gian lớp ẩn tương tự tương tự như điểm dữ liệu bình thường. Đó có thể là lý do SAE sẽ gặp khó cho phân tách mẫu dữ liệu bình thường và bất thường trong trong trường hợp nêu trên. Nhận định trên cũng phù hợp với số liệu từ kết quả công bố của tác giả đã đề xuất giải pháp SAE [20, xem Bảng 3], số liệu cho thấy, SAE gặp cho hiệu quả không tốt với loại tấn công Remote to Local (R2L), đây được cho là loại tấn công mạng nguy hiểm và khác so với đa số tấn công mạng khác như DoS hay Probe [71]. Tấn công mạng R2L nhúng bản thân mã độc trong các gói tin dữ liệu và không tạo ra các mẫu tuần tự như tấn công DoS và Probe. Điều này làm cho R2L có lưu lượng mạng gần giống với dữ liệu bình thường [3], [56], [71].

Khi xem xét hai vấn đề trên theo chiều xử lý của dữ liệu của phương pháp SAE có thể nhận thấy, hạn chế thứ nhất nằm ở việc vấn đề xử lý dữ liệu trước khi đẩy vào SAE, ngược lại hạn chế thứ hai hoàn toàn nằm trong phần lõi SAE, việc xử lý cần phải được cải tiến nội tại trong SAE. Do vậy, hai hạn chế này hoàn toàn độc lập và có thể nghiên cứu riêng, kết quả xử lý từng hạn chế đều góp phần cải tiến thuật toán SAE hiện có.

2.2. Giải pháp đề xuất

Như đã phân tích ở phần Giới thiệu, hai vấn đề mà SAE có thể đang gặp phải nằm ở các giai đoạn khác nhau của mô hình SAE, do vậy để dễ dàng cho việc mô tả kết quả cải tiến, phát triển. Để có thể dễ hơn trong việc so sánh, đánh giá các đề xuất cải tiến, Luận án tách giải pháp xử lý riêng biệt cho hai hạn

chế đặt ra đối với SAE. Đầu tiên là cải tiến SAE bằng giải pháp có tên KSAE, thực hiện ở giai đoạn xử lý dữ liệu trước khi đẩy vào SAE. Tiếp đó, phát triển lõi của SAE thông qua đề xuất giải pháp có tên DSAE. Mô tả mối liên hệ trên SAE, KSAE và DSAE như trên Hình 2.2, trong đó x_i là mẫu dữ liệu đầu vào, S_i là độ đo bất thường tại đầu ra.



Hình 2.2: *Minh họa mối liên hệ SAE, KSAE và DSAE*

2.2.1. Giải pháp Clustering-Shrink AutoEncoder

Để khắc phục hạn chế thứ nhất của SAE, Luận án đề xuất giải pháp kết hợp kỹ thuật phân cụm và SAE, đặt tên là KSAE (Clustering-Shrink AutoEncoder). Dựa trên giả định rằng, phiên bản gốc của SAE được huấn luyện để điều hướng toàn bộ dữ liệu bình thường về gốc toạ độ trong không gian dữ liệu lớp ẩn trung tâm của AE. Vì vậy, khi gặp dữ liệu đã tồn tại ở dạng nhiều cụm thì SAE có thể hoạt động không hiệu quả.

Phân cụm là chia dữ liệu thành các nhóm đối tượng tương đương [11], việc chia thành nhiều cụm để giúp giảm kích thước dữ liệu mà vẫn giữ được đặc trưng của dữ liệu, dữ liệu lúc này được mô tả bằng từng cụm riêng lẻ. Trong lĩnh vực học máy, phân cụm thuộc bài toán học không giám sát, mục tiêu của mô hình

phân cụm là gán nhãn cho dữ liệu theo số cụm cho trước hoặc số cụm tối ưu nhất có thể theo từng bài toán. Thuật toán phổ biến nhất cho phân cụm có thể kể đến là K-means clustering (K-means) được đề xuất bởi Mac Queen [11]. Nhờ sự đơn giản, hiệu quả mà K-means được ứng dụng nhiều trong lĩnh vực khai phá dữ liệu. Về bản chất, phương pháp đề xuất có thể hoạt động với mọi thuật toán phân cụm, tuy nhiên để tiện cho mô tả giải pháp và cài đặt thực nghiệm, luận án chọn K-means đại diện cho bước phân cụm trong mô hình học sâu KSAE. K-means hoạt động trên cơ sở, từ tập dữ liệu với N điểm, thuật toán thực hiện trên cơ sở xác định K trung tâm là đại diện cho K cụm dữ liệu được tạo ra, K trung tâm được xác định dựa vào trung bình khoảng cách của các điểm tương ứng thuộc cụm đó đến các trung tâm.

Công đoạn chia thành K cụm cho trước được thực hiện trước khi áp dụng SAE. Theo đó, quá trình huấn luyện mô hình KSAE gồm hai công đoạn: Thứ nhất, dữ liệu đầu vào được phân cụm sử dụng thuật toán phân cụm (TTPC), thuật toán này được huấn luyện để chia tập dữ liệu theo số cụm K, cho trước. Thứ hai, ứng với số cụm K được chia tách, các mô hình SAE được huấn luyện bởi chỉ dữ liệu ứng với cụm dữ liệu tương ứng thu được từ bước thứ nhất. Thuật toán 2.1 trình bày chi tiết quá trình huấn luyện của KSAE.

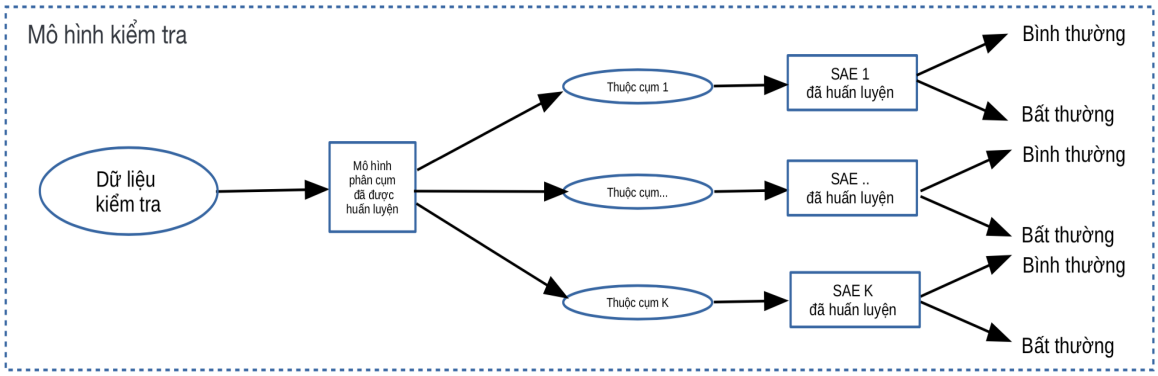
Thuật toán 2.1 Huấn luyện mô hình KSAE

INPUT: Tập huấn luyện D_n, số cụm cho trước K.
OUTPUT: $trained^{TTPC}$, K $trained^{SAE}$.

1: $trained^{TTPC} \leftarrow$ huấn luyện thuật toán phân cụm với đầu vào D_n, K.
2: K tập huấn luyện $D^j \leftarrow$ kiểm tra $trained^{TTPC}$ với đầu vào D_n.
3: $j \leftarrow 0$.
4: **while** $j < K$ **do**
5: $\quad trained_j^{SAE} \leftarrow$ huấn luyện SAE với tập dữ liệu D^j.
6: **end while**
7: Trả về $trained^{TTPC}$, K $trained^{SAE}$.

Sau khi huấn luyện, chúng ta thu được 1 (một) mô hình phân cụm (K-means) và K mô hình SAE đã được huấn luyện. Các mô hình này sau đó được sử dụng cho quá trình kiểm tra.

Mô hình kiểm tra KSAE như tại Hình 2.3, trong mô hình kiểm tra này, các mẫu dữ liệu đầu vào đầu tiên được kiểm tra để xác định số cụm bởi mô hình phân cụm đã được huấn luyện, kết quả trả về là nhãn $C_j \leq K$, ứng với cụm của dữ liệu đầu vào. Mô hình SAE_j tương ứng sau đó được sử dụng cho kiểm tra để xác định độ đo bất thường ứng với điểm dữ liệu đầu vào.

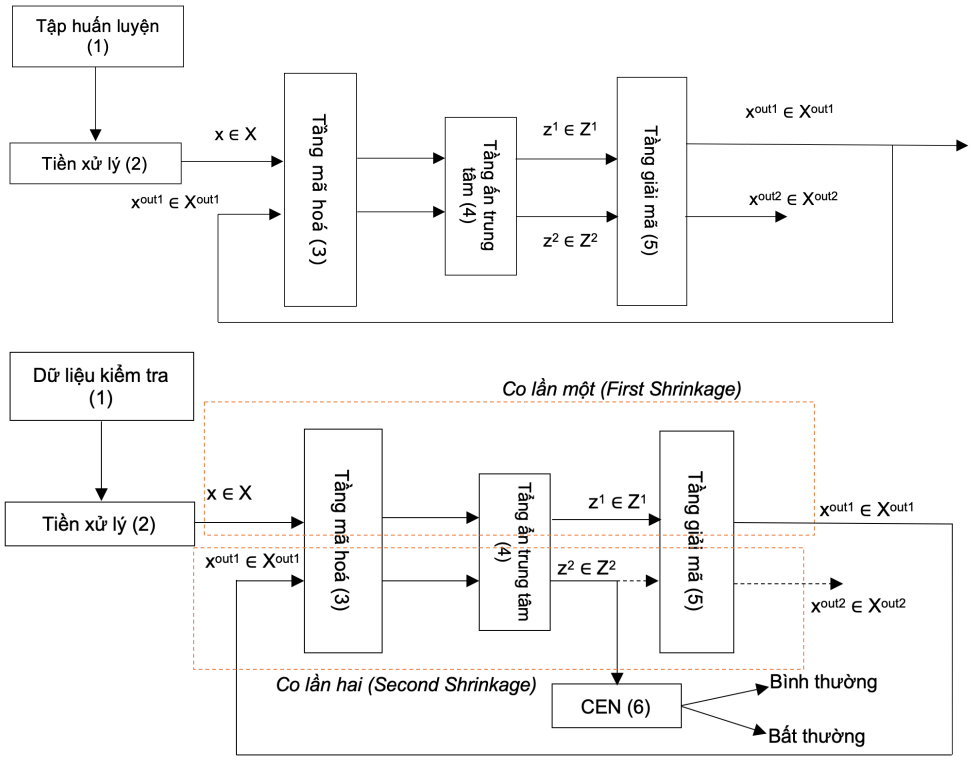


Hình 2.3: *Mô hình kiểm tra theo phương pháp KSAE*

2.2.2. Giải pháp Double-shrink AutoEncoder

Với hạn chế, SAE gặp khó khăn với một số loại tấn công nhất định, luận án đưa ra giải pháp cải tiến nhân của SAE, giải pháp có tên DSAE (Double-Shrink AutoEncoder), nội dung được trình bày như sau. Lỗi tái tạo (RE) của mô hình dựa trên AutoEncoder có thể thể hiện mức độ bất thường của dữ liệu, RE lớn thể hiện việc tái tạo dữ liệu hạn chế, dữ liệu có tính bất thường cao và ngược lại. Với các bất thường mà SAE gặp khó, có hai trường hợp cho vector tái tạo đầu ra của SAE trong trường hợp này. Thứ nhất, cho lỗi tái tạo (RE) nhỏ, khi đó mẫu bất thường đầu ra (được tái tạo) sẽ gần giống với mẫu bất thường đầu vào, và là bất thường. So với mẫu bất thường đầu vào, mẫu dữ liệu được tái tạo này có thể khác xa hơn mẫu dữ liệu bình thường. Điều này có thể giải thích, vì mẫu đầu vào là bất thường, nên qua mạng nơ-ron AE sẽ cho giá trị tái tạo tại đầu ra (X-out), cũng là bất thường. X-out có xu thế khác xa với mẫu bình thường hơn vì nó đã qua thêm lần được tái tạo lại từ mẫu bất thường. Thứ hai, nếu RE lớn

thì mẫu bất thường được tái tạo, X-out, có xu thế khác xa hơn so với mẫu bất thường đầu vào, nghĩa là khác xa hơn so với mẫu dữ liệu bình thường. Từ phân tích trên, ý tưởng là sử dụng thêm dữ liệu của vector tái tạo đầu ra, X-out, để cho việc phân tách dữ liệu bình thường và bất thường hiệu quả hơn, cụ thể là sử dụng dữ liệu tái tạo đầu ra của SAE một lần nữa theo cách mà SAE đã làm để thu được vector lớp ẩn. Và ở lần co (shrinkage) này, từ mẫu dữ liệu bất thường X-out có nhiều khả năng để tạo ra vector lớp ẩn ở vị trí xa gốc toạ độ hơn so với lần co thứ nhất. Do vậy mô hình đề xuất có thể phân tách được tốt hơn đối với dữ liệu bất thường mà SAE gặp khó.



Hình 2.4: *Mô hình Double-shrink AutoEncoder*

Hình 2.4 mô tả quá trình huấn luyện và kiểm tra mô hình DSAE. Chỉ có dữ liệu bình thường được sử dụng cho huấn luyện DSAE, dữ liệu tại lớp đầu ra tiếp tục được sử dụng như đầu vào lần thứ hai cho mô hình DSAE. Mô hình được huấn luyện để đồng thời đạt được các mục tiêu gồm giảm thiểu lỗi tái tạo đầu ra và ràng buộc để các vector lớp ẩn trung tâm (z^1 và z^2, thuộc không gian lớp

ẩn như trên Hình 2.4, $Z^1 = Z^2 = R^l$, trong đó l là kích thước lớp ẩn) hướng về gốc toạ độ. Bởi vậy, các bất thường mà SAE gặp khó, được giả định là có dữ liệu rất giống với bình thường, sau khi được thực hiện co lại (shrinkage) lần thứ nhất sẽ tạo ra các giá trị vector z^1 gần với gốc toạ độ với giá trị lỗi tái tạo tương ứng RE thường có thể rất bé, nhưng thực tế giá trị tái tạo thu được, x^{out1}, vẫn là dữ liệu bất thường. Do vậy, với lần co tiếp theo sẽ dẫn đến tạo ra vector z^2 có xu hướng bị đẩy ra xa gốc toạ độ hơn so với z^1. Trong quá trình huấn luyện, dữ liệu bình thường được đẩy vào mô hình DSAE như với SAE, mục tiêu huấn luyện hướng tới giảm thiểu lỗi tái tạo lần co 1 (RE_1) và lần co 2 (RE_2) đồng thời điều hướng các vector z^1 và z^2 về gốc toạ độ trong không gian lớp ẩn. Trong đó $z^1 = f_\theta(x)$, là vector lớp ẩn trung tâm cho lần co thứ nhất; $z^2 = f_\theta\left(g_\theta\left(z^1\right)\right)$, là vector lớp ẩn cho lần co thứ hai. Quá trình kiểm tra, z^1, z^2 được sử dụng như vector đặc trưng đại diện cho dữ liệu đầu vào gốc. Các dữ liệu đầu ra tại lớp ẩn trung tâm này có thể được sử dụng trực tiếp để tính độ đo bất thường (thông qua khoảng cách Euclid từ véc tơ đến gốc toạ độ) hoặc được đẩy vào một thuật toán phát hiện bất thường bất kỳ (ví dụ như CEN [17], là phương pháp dễ sử dụng), để cho kết quả cuối cùng là một độ đo bất thường. Mô hình DSAE mặc định sử dụng vector z^2 cho biểu diễn dữ liệu đầu vào cho mục đích phát hiện bất thường, cũng có thể được ký hiệu là DSAE_Z2.

Khi xem xét giải pháp đề xuất trên phương diện toán học, cụ thể là hàm mất mát (thường gọi là Loss Function hay Cost Function) có thể được trình bày như sau. Các giá trị lỗi tái tạo RE_1 và RE_2 có thể được định nghĩa:

$$L_{RE_1}(\theta, x_i) = \frac{1}{m}\sum_{i=1}^{m}\left(x_i - x_i^{out1}\right)^2 \tag{2.1}$$

$$L_{RE_2}(\theta, x_i) = \frac{1}{m}\sum_{i=1}^{m}\left(x_i^{out1} - x_i^{out2}\right)^2 \tag{2.2}$$

trong đó θ là bộ tham số cho DSAE, m là số mẫu dữ liệu cho huấn luyện, còn

x_i, x_i^{out1} và x_i^{out2} là dữ liệu đầu vào thứ i và giá trị đầu ra tương ứng với lần co thứ nhất và thứ hai.

Theo đó, thành phần REs của hàm mất mát theo phương pháp DSAE có thể trình bày theo Công thức 2.3.

$$L_{RE}(\theta, x_i) = L_{RE_1}(\theta, x_i) + L_{RE_2}(\theta, x_i) \tag{2.3}$$

Còn thành phần điều chuẩn co trong hàm mất mát của phương pháp DSAE là tổng của hai lần co và được biểu diễn bởi Công thức 2.4, 2.5 và 2.6:

$$L_{Z_1}(\theta, x_i) = \frac{1}{m}\sum_{i=1}^{m}\left\|z_i^1\right\|^2 \tag{2.4}$$

$$L_{Z_2}(\theta, x_i) = \frac{1}{m}\sum_{i=1}^{m}\left\|z_i^2\right\|^2 \tag{2.5}$$

$$L_Z(\theta, x_i) = L_{Z_1}(\theta, x_i) + L_{Z_2}(\theta, x_i) \tag{2.6}$$

trong đó z_i^1 và $z_i 2$ là các vector ẩn của dữ liệu đầu vào x_i tương ứng tại lần co thứ nhất và thứ hai.

Từ trình bày trên, hàm mất mát của DSAE được viết lại như sau:

$$L_{DSAE}(\theta, x_i) = L_{RE}(\theta, x_i) + \alpha * L_Z(\theta, x_i) + \beta * L_W(\theta, x_i) \tag{2.7}$$

Thành phần cuối cùng của hàm mất mát để kiểm soát việc suy giảm trọng số (weight decay regularizer) cho bộ trọng số của mạng nơ-ron, W, được trình bày trong Công thức 2.8 dưới đây:

$$L_W(\theta, x_i) = \sum_{l=1}^{L}\left\|W^l\right\|_F^2 \tag{2.8}$$

trong đó $\|.\|_F$ là chuẩn hoá Frobenius [17]; các hệ số α và β điều khiển sự cân bằng giữa ba thành phần của hàm mất mát DSAE.

2.3. Thực nghiệm

Phần này mô tả về việc triển khai thực nghiệm, bao gồm các bộ dữ liệu và thiết lập tham số cho các thực nghiệm.

2.3.1. Dữ liệu thực nghiệm

Với mục đích kiểm thử để đánh giá các phương pháp, thuật toán đề ra. Quá trình thực nghiệm sử dụng các bộ dữ liệu phổ biến và hiện đại trong lĩnh vực an ninh mạng, các bộ dữ liệu này đã được giới thiệu tại phần 1.4.1. Bao gồm, sử dụng 04 bộ dữ liệu thuộc tập CTU13 [42] được công bố năm 2014, bộ dữ liệu mạng thực UNSW-NB15 [75] được công bố năm 2015 và bộ dữ liệu NSL-KDD [97] được công bố năm 2009. Trong thực nghiệm, tất cả các loại tấn công đều được xem là bất thường (Anomaly), còn lại là dữ liệu bình thường (Normal), chi tiết tại Bảng 2.1.

Bảng 2.1: *Các bộ dữ liệu sử dụng cho thực nghiệm*

Bộ dữ liệu	**Số chiều nguyên bản/ sau one-hot encoding**	**Tập huấn luyện**	**Tập kiểm tra**	
			Bình thường	***Bất thường***
NSLKDD	44/122	67343	9711	12833
UNSW-NB15	47/196	56000	37000	45332
CTU13_08	16/40	29128	43694	3677
CTU13_09	16/41	11986	17981	110998
CTU13_10	16/38	6338	9509	63812
CTU13_13	16/40	12775	19164	24002

Với các thực nghiệm để so sánh hiệu quả của các mô hình trên các nhóm tấn công mạng khác nhau. Luận án chọn cách phân tách tấn công mạng thành bốn nhóm, bao gồm *Từ chối dịch vụ (Denial of service - DoS)*, *Từ xa vào nội bộ (Remote to Local - R2L)*, *Leo thang đặc quyền (User to Local - U2R)*, và *Dò quyét (Probe)* [3], [56]. Lý do vì cách phân nhóm này thể hiện các nhóm tấn công rất khác nhau, đã được chấp nhận rộng rãi từ lâu. Thực nghiệm sử dụng

tập dữ liệu NSL-KDD, được đánh giá là phù hợp cho các nghiên cứu mới trong lĩnh vực an ninh mạng và học máy [13], [17] [35], và được biết là tập dữ liệu mới nhất được phân tách theo bốn nhóm tấn công trên. Trong số 4 nhóm tấn công trên, R2L được xem là loại tấn công khó được phát hiện bởi các thuật toán học máy [71]. R2L hoạt động dựa trên ẩn nội dung của nó trong các gói tin, do vậy dữ liệu tạo ra không giống với các loại tấn công phổ biến khác như DoS và Probe. Đó có thể là nguyên nhân chính dẫn đến dữ liệu lưu lượng mạng hình thành từ tấn công R2L có thuộc tính khá tương tự với các lưu lượng mạng bình thường khác [3], [56], [71]. Với tập dữ liệu nhóm tấn công R2L, có 995 mẫu trong tập *KDDTrain+*, làm tập huấn luyện và 2887 mẫu trong tập *KDDTest+*, làm tập kiểm tra.

2.3.2. Phương pháp xác định số cụm tối ưu

Trong ứng dụng thuật toán phân cụm, việc dữ liệu có nên phân thành cụm nhỏ hơn hay không và nên chia thành bao nhiêu cụm dữ liệu là một vấn đề cần phải giải quyết. Giải pháp yêu cầu trả lời được câu hỏi, tập dữ liệu có tính phân cụm không, số cụm tối ưu K nên phân ra là bao nhiêu. Đây là tham số đầu vào cho mô hình KSAE như đã mô tả tại Thuật toán 2.1. Có nhiều phương pháp để xác định số cụm tối ưu, phổ biến trong đó là Elbow, cho phép xác định số K tối ưu dựa vào trực quan trên biểu đồ. Theo phương pháp khủy tay (Elbow), một đồ thị 2D sẽ được biểu diễn bởi trục hoành là số cụm dự kiến sẽ chia (ví dụ từ 1-5), trục tung biểu diễn bằng tổng bình phương khoảng cách (Within-cluster Sum of Square - WSS) tất cả các điểm đến trung tâm cụm C_j. Số K tối ưu được xác định ứng với điểm tại đó trục hoành và đồ thị tạo nên khủy tay, Công thức cho xác định WSS theo như sau,

$$WSS_k = \sum_{r=1}^{k} \frac{1}{n_r} D_r \tag{2.9}$$

trong đó k là số cụm, n_r là số điểm dữ liệu trong cụm r, D_r là tổng số khoảng cách giữa tất cả các điểm trong một cụm, được tính theo Công thức 2.10 sau,

$$D_r = \sum_{i=1}^{n_r-1} \sum_{j=1}^{n_r} \|x_i - x_j\|_2 \tag{2.10}$$

2.3.3. Thiết lập tham số thực nghiệm

Thuật toán K-means được sử dụng như thuật toán phân cụm cho các thực nghiệm của mô hình KSAE. Các tham số mạng nơ-ron được chọn theo [20], số lượng lớp ẩn cho các mạng nơ-ron là 5, kích thước lớp ẩn trung tâm, l, được chọn theo khuyến nghị tại [18], $l = [1 + \sqrt{d}]$, trong đó d là số thuộc tính đầu vào. Tập batch (kích thước nhóm huấn luyện) có kích thước 100, sử dụng hàm kích hoạt tanh ngược (hyperbolic tangent) cho tất cả các lớp. Trọng số mạng nơ-ron được khởi tạo theo phương pháp Xavier [46]. Trong quá trình tối ưu bằng thuật toán đạo hàm lặp ADADELTA [115]. Mỗi mô hình được huấn luyện trong 1000 chu kỳ (epochs), kỹ thuật dừng sớm huấn luyện (early stopping) được thực hiện nếu không có bất có cải tiến trong giảm giá trị hàm mất mát sau 10(mười) epochs. Trong thực nghiệm, mô hình DSAE sử dụng $\alpha = 10$ như được khuyến nghị bởi Cao et al. [20], và $\beta = 0.001$ là giá trị phổ biến thường được chọn trong các nghiên cứu [64].

Việc cài đặt thực nghiệm được tiến hành trên ngôn ngữ Python 3.0, công cụ phát triển Jupyter Notebook, nền tảng Tensorflow cho cài đặt mạng nơ-ron, các thư viện Sklearn, Numpy và pandas cho cài đặt các thuật toán. Kết quả thực nghiệm thực hiện trên máy tính có hệ điều hành MAC OS 10.14.3, cấu hình: Intel(R) Core (TM) i5, 8GB DDR3.

2.4. Kết quả và đánh giá

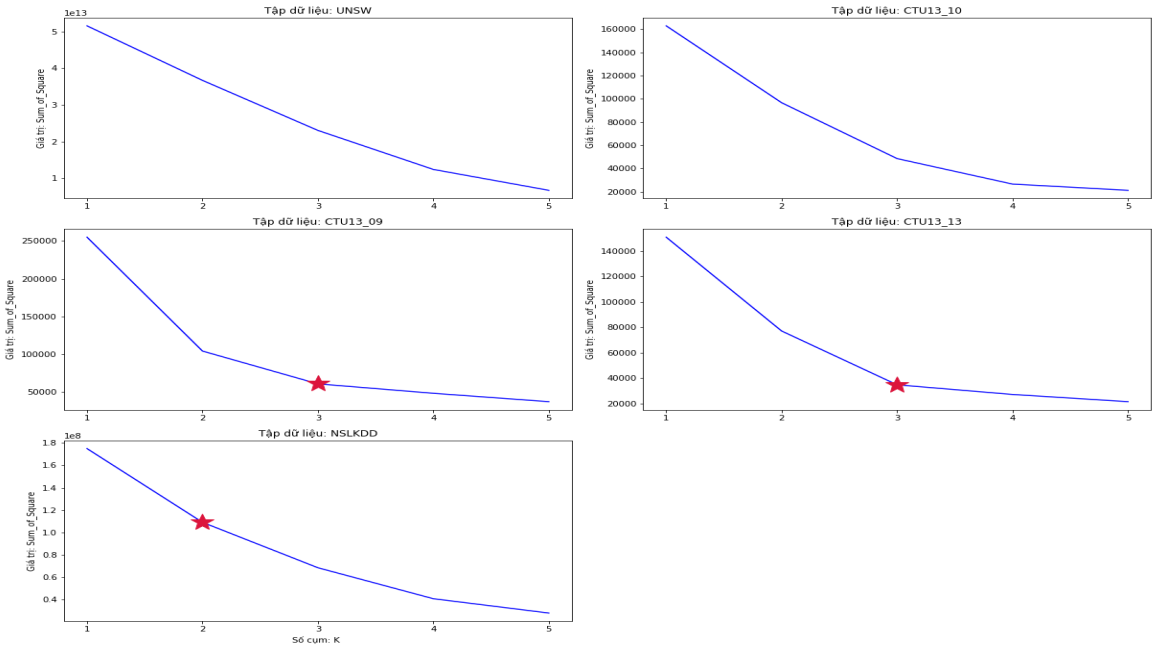
Đánh giá dựa trên kết quả thực nghiệm KSAE: Với giả định đặt ra, mô hình mạng nơ-ron học sâu tiêu biểu, SAE, hoạt động không hiệu quả trên dữ liệu hiện hữu nhiều cụm. Luận án đề xuất khắc phục trên cơ sở đề xuất thuật toán kết hợp một kỹ thuật phân cụm và SAE, gọi là KSAE như đã trình bày tại phần 2.2. Việc thực nghiệm đánh giá được thực hiện theo hai bước. Bước đầu tiên là để so sánh hiệu quả của phương pháp đề xuất với với mô hình mạng nơ-ron học sâu tiêu biểu hiện tại, SAE [20]. Sau đó, phân tích tính phân cụm hiện có của các dữ liệu thực nghiệm để đối chiếu lại kết quả thực nghiệm ở bước trên. Với thực nghiệm thứ nhất, vì chưa có cơ sở xác định ngưỡng quyết định do vậy chỉ số AUC (Area Under the Curve) được sử dụng để so sánh khả năng phát hiện bất thường giữa hai mô hình. Các bộ dữ liệu được chia thành số cụm $K = (1, 2, 3)$ khi thực nghiệm. Một điểm lưu ý rằng, với trường hợp K=1, lúc này mô hình KSAE hoàn toàn đồng nhất với SAE. Các mô hình sử dụng CEN [17] để tính độ đo bất thường. Bản chất CEN cung cấp độ đo bất thường, là giá trị tương ứng với khoảng cách từ vector đầu vào tới gốc toạ độ.

Bảng 2.2: *Kết quả AUC của KSAE trên các tập dữ liệu*

Số cụm K	**Tập dữ liệu**				
	NSL-KDD	**UNSW**	**CTU13-09**	**CTU13-10**	**CTU13-13**
K=1	0.941	**0.887**	0.923	**0.998**	0.931
K=2	**0.962**	0.885	0.935	0.989	0.933
K=3	0.879	0.858	**0.946**	0.965	**0.962**

Có thể nhận thấy rằng, với một giá trị K phù hợp, mô hình đề xuất có khả năng phát hiện cải tiến so với SAE. Cụ thể kết quả ứng với mô hình kiểm tra

như tại Bảng 2.2 cho thấy, mô hình đề xuất có kết quả tốt hơn trên 3 tập dữ liệu. Ví dụ, với NSL-KDD, AUC cho KSAE tăng từ 0.941 đến 0.962 với K= 2. Tuy vậy, trên hai bộ dữ liệu còn lại (UNSW-NB15 và CTU13_10), hiệu năng của KSAE không tốt bằng SAE. Điều này cũng không ngạc nhiên vì kết quả trong bước thực nghiệm thứ hai cho thấy, UNSW-NB15 và CTU13-10 tồn tại ở dạng một cụm. Điều này giải thích vì sao, khi sử dụng thuật toán phân cụm kết hợp SAE không giúp cho cải thiện độ chính xác của SAE trong phát hiện bất thường.



Hình 2.5: *Kết quả phương pháp Elbow trên các tập dữ liệu.*

Thực nghiệm thứ hai là để kiểm tra lại tính phân cụm của các tập dữ liệu được sử dụng cho thực nghiệm, với mỗi bộ dữ liệu thì số cụm tối ưu K được phân là bao nhiêu. Sử dụng kỹ thuật Elbow để ước lượng số cụm tối ưu mà một bộ dữ liệu nên tách ra trước khi ứng dụng SAE. Kết quả của Elbow sẽ đối chiếu với thực tế thực nghiệm tại bước một để đánh giá tính đồng nhất.

Kết quả thực nghiệm theo phương pháp Elbow trên 5 bộ dữ liệu, qua 5 lần thử, mỗi lần thử tính K = (1 đến 5) trên tập lấy mẫu ngẫu nhiên 10% dữ liệu của tập huấn luyện NSL-KDD, UNSW-NB15 và 20% trên các bộ dữ liệu thuộc

nhóm CTU13, kết quả cho thấy độ ổn định ở các lần thử khác nhau. Sơ đồ thể hiện vị trí Elbow khi thực nghiệm đối với các bộ dữ liệu cho lần thử thứ nhất thể hiện tại Hình 2.5. Theo đó với bộ dữ liệu UNSW-NB15, việc tách bộ dữ liệu thành K cụm khác nhau (với K từ 1 đến 5) đều thể hiện không rõ bởi phương pháp Elbow, điều này có vẻ đồng nhất với kết quả tại bước thực nghiệm thứ nhất như trong Bảng 2.2. Đó là bộ dữ liệu UNSW-NB15 có thể hiện hữu tốt nhất trong một cụm duy nhất, và lý giải cho vấn đề tại sao với bộ dữ liệu UNSW-NB15 thì KSAE không tốt hơn SAE.

Còn với các bộ dữ liệu còn lại, CTU13_9, CTU13_13 đều thể hiện rất rõ Elbow tại vị trí K= 3, còn NLS-KDD thể hiện Elbow ở K= 2. Kết quả này cũng hỗ trợ cho kết quả tại bước thực nghiệm thứ nhất như Bảng 2.2. Riêng với bộ dữ liệu CTU13_10 đường cong Elbow thể hiện thay đổi rõ nét tại K= 3 và K= 4 tuy nhiên hiệu năng của KSAE lại không tốt hơn SAE tại các vị trí K này. Vấn đề có thể xuất phát từ sự ảnh hưởng của độ phân mảnh (sparity) tới sự đồng nhất kết quả của Elbwo và KSAE, vì CTU13_10 có độ phân mảnh 0.71, còn UNSW-NB15 và NSL-KDD tương ứng là 0.84 và 0.88.

Tổng quan lại, mô hình đề xuất của KSAE cho thấy khả năng cải tiến hiệu quả phát hiện bất thường so với mô hình SAE khi hoạt động với các bộ dữ liệu hiện hữu nhiều cụm. Thêm vào đó, chúng ta có thể sử dụng các phương pháp như Elbow để đánh giá, tính toán số cụm hiện hữu trong mỗi tập dữ liệu được quan sát. Kết quả trên cho thấy phương pháp kết hợp phân cụm và SAE có thể giúp cho mô hình mạng nơ-ron học sâu có thể khắc phục hạn chế khi làm việc với dữ liệu hiện hữu trong nhiều cụm.

Đánh giá dựa trên kết quả thực nghiệm DSAE:

Với giả định đặt ra, mô hình mạng nơ-ron học sâu tiêu biểu, mô hình SAE, gặp khó khăn với một số loại tấn công, Luận án đề xuất giải pháp phát triển tử SAE có tên DSAE như đã được trình bày trong 2.2. Để kiểm chứng giải pháp, quá trình thực nghiệm được tiến hành theo hai bước. Thực nghiệm thứ nhất để so sánh khả năng phát hiện bất thường của DSAE với các mô hình

tiên tiến NAD sử dụng mạng nơ-ron học sâu, bao gồm SAE [20] và Denoising AutoEncoder (DAE) [109]. Thực nghiệm thứ hai để đánh giá mức độ hiệu quả của hai mô hình DSAE và SAE trên các nhóm tấn công mạng có thể SAE gặp khó.

Để đánh giá hiệu quả giữa các mô hình NAD, chỉ số AUC (Area Under the ROC Curve) được sử dụng. Giá trị AUC lớn hơn chứng tỏ mô hình có khả năng phát hiện bất thường tốt hơn. Ngoài ra, khi đánh giá mô hình ở các ngưỡng cụ thể, các chỉ số TP, FP, FN, TN và cặp chỉ số DR, FAR được sử dụng để so sánh tính hiệu quả các mô hình đối với nhóm tấn công cụ thể.

Bảng 2.3: *AUC từ các mô hình DAE, SAE, DSAE trên sáu tập dữ liệu*

Phương pháp	**Tập dữ liệu**					
	NSLKDD	**UNSW**	**CTU13-08**	**CTU13-09**	**CTU13-10**	**CTU13-13**
DAE* + CEN	0.854 ±0.002	0.690 ±0.001	0.938 ±0.015	0.655±0.031	0.951±0.006	0.711±0.002
DAE+RE	0.930±0.090	0.873±0.004	0.960±0.011	0.903±0.002	0.958±0.004	0.952±0.010
SAE**+CEN	0.960 ±0.002	0.896 ±0.006	0.982 ±0.009	0.940 ±0.010	0.997 ±0.001	0.964 ±0.012
SAE+RE	0.920 ±0.000	0.810 ±0.001	0.951 ±0.013	0.703 ±0.020	0.997 ±0.000	0.887 ±0.005
DSAE + CEN	**0.963** ±0.004	0.895 ±0.015	**0.986** ±0.012	0.929 ±0.054	0.992 ±0.008	**0.971** ±0.006

* DAE: Denoising AutoEncoder; ** SAE: Shrink AutoEncoder [20]

Trong thực nghiệm thứ nhất để so sánh hiệu năng của DSAE với SAE và DAE trên 6 bộ dữ liệu phổ biến, nổi tiếng trong lĩnh vực an ninh mạng. Kết quả được trình bày tại Bảng 2.3, với DAE và SAE đều trình bày hai phiên bản kết quả. Phiên bản thứ nhất, (DAE+RE và SAE+RE) sử dụng RE như là đơn vị đo độ bất thường. Phiên bản còn lại, sử dụng CEN trên vector lớp ẩn của DAE và SAE. Với DSAE, chỉ trình bày kết quả DSAE+CEN, vì kết quả DSAE+RE không nhiều ý nghĩa cho so sánh. Từ Bảng 2.3 cho thấy AUC của SAE và DSAE tương đương nhau trên hầu hết các tập dữ liệu được kiểm thử, và giá trị này tốt hơn DAE. Kết quả này khẳng định rằng, mô hình đề xuất có khả năng phát hiện bất thường so sánh được với mô hình SAE, và hiệu quả hơn DAE.

Kết quả thực nghiệm thứ hai cho thấy như sau. Bảng 2.4 trình bày kết quả

Bảng 2.4: *AUC từ SAE, DSAE trên bốn nhóm tấn công tập dữ liệu NSL-KDD*

Phương pháp	**Tập dữu liệu**			
	Probe	**DoS**	**R2L**	**U2R**
SAE + CEN *	0.977 ±0.003	0.967 ±0.002	0.924 ±0.010	0.956 ±0.005
DSAE + CEN	0.979 ±0.006	0.966 ±0.007	**0.936** ±0.011	0.960 ±0.010

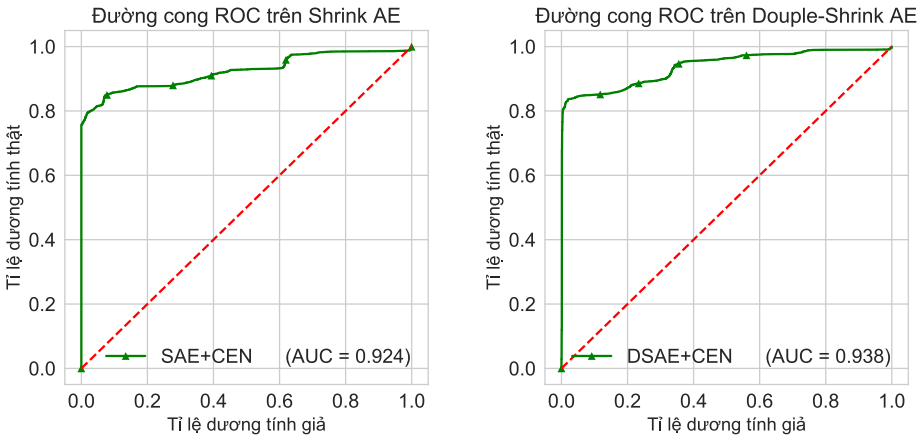
* SAE + CEN: Shrink AutoEncoder và Centroid [20]

Bảng 2.5: *Kết quả DR, FAR giữa SAE và DSAE trên nhóm tấn công R2L*

Phương pháp	**Dữ liệu nhóm tấn công R2L**					
	TP	**FP**	**FN**	**TN**	**FAR**	**DETECTION RATE**
SAE + CEN *	1892	1008	995	8702	0.104	0.655
DSAE + CEN	2011	989	876	8721	**0.102**	**0.697**

* SAE + CEN: Shrink AutoEncoder và Centroid [20]

DSAE và SAE trên bốn nhóm tấn công của NSL-KDD. Số liệu chứng tỏ DSAE cho khả năng phát hiện so sánh được với SAE trên ba nhóm tấn công (Probe, DoS, và U2R). Tuy nhiên, trên nhóm tấn công khó nhất là R2L, DSAE (AUC ≈ 0.936) cho kết quả ấn tượng hơn với SAE (AUC ≈ 0.924). Khi xem xét đường ROC của hai mô hình như tại Hình 2.6 cũng cho thấy, đỉnh của đường cong ROC theo mô hình DSAE hướng gần hơn tới đỉnh (0,1) so với đỉnh đường cong ROC của mô hình SAE.



Hình 2.6: *Giá trị AUC của SAE, DSAE trên nhóm tấn công R2L*

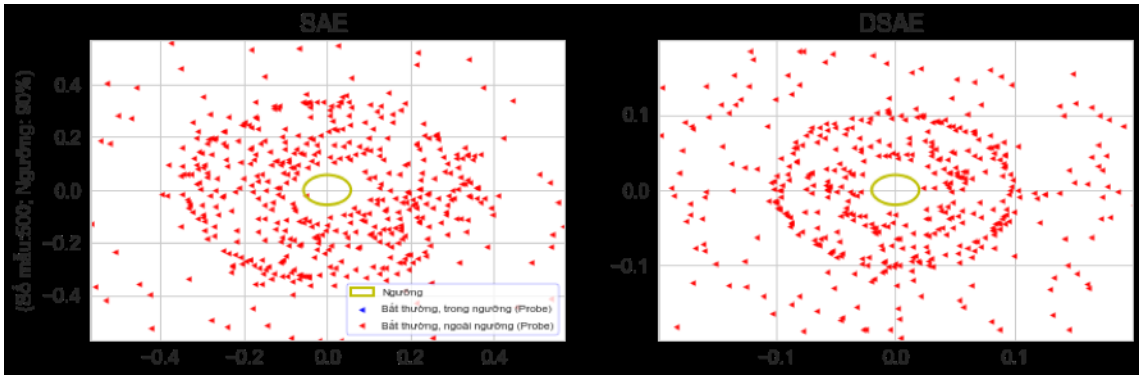
Kết quả này thể hiện rằng DSAE có thể cải thiện được khả năng phát hiện đối với các tấn công mạng như R2L, loại tấn công hoạt động dựa trên ẩn nội dung của nó trong các gói tin và do vậy cho dữ liệu tương tự với các lưu lượng mạng bình thường khác [3], [56], [71].

Tính hiệu quả của DSAE so với SAE trên các nhóm tấn công khác nhau có thể được thể hiện thông qua sự chuyển dịch của các vector lớp ẩn tương ứng so với gốc toạ độ và giá trị ngưỡng quyết định. Ngưỡng là giá trị ứng với khoảng cách Euclid từ điểm dữ liệu đến gốc toạ độ trong không gian lớp ẩn, tại đó tương ứng số phần trăm ($t\%$) mẫu dữ liệu của tập huấn luyện khi tham gia kiểm tra bé hơn giá trị này, nghĩa là dự đoán "bình thường". Nhìn chung t thường được chọn trong khoảng $90 - 97\%$ [17]. Số liệu trình bày trong thực nghiệm có ($t\%$) tương ứng với 90%, theo như [20].

Như đã đề cập, các tấn công SAE gặp khó là tấn công cho vectơ lớp ẩn trong kiến trúc AE gần gốc toạ độ (bé hơn ngưỡng) với giá trị sai số tái tạo RE có thể là bé hoặc lớn hơn ngưỡng phân tách tương ứng. Bảng 2.6 cho thấy, với các tấn công cho "RE lớn" mà SAE phân tách lỗi thì đều được DSAE phân tách đúng (26/26 mẫu với R2L). Còn đa số mẫu tấn công khó với SAE mà cho "RE bé" thì được DSAE phân tách đúng, cụ thể số mẫu đã phân tách đúng/số có RE

Bảng 2.6: *Kết quả DSAE phân tách các nhóm tấn công SAE có thể gặp khó*

Nhóm tấn công	Tấn công SAE gặp khó đúng			Các tấn công SAE khó, được DSAE phân tách đúng	
	Tổng	**RE bé**	**RE lớn**	**Re bé**	**RE lớn**
Probe	0	0	0	0	0
DoS	434	434	0	327	0
R2L	146	120	26	95	26
U2R	6	6	0	3	0
Tổng	586	560	26	418	26



Hình 2.7: *Không gian lớp ẩn nhóm tấn công Probe trên SAE, DSAE*

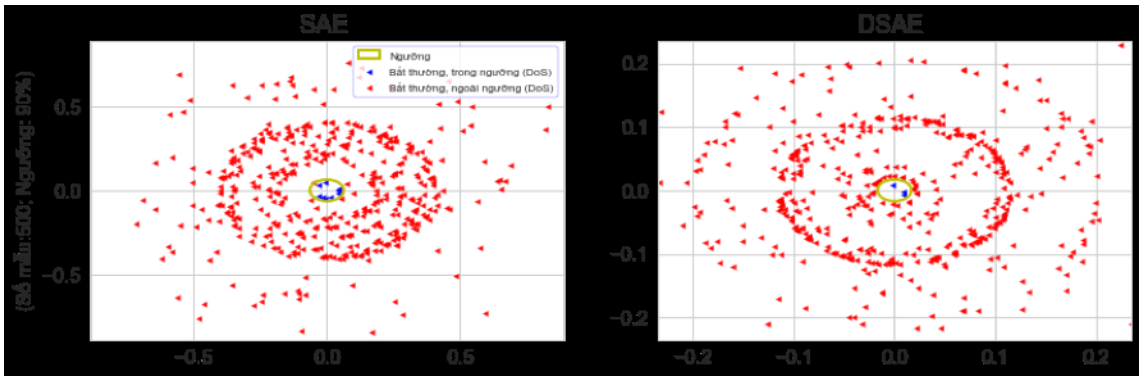
bé là: 95/120 với R2L; 3/6 với U2R; 327/434 với DoS. Riêng Probe không cho thấy có mẫu khó với mô hình SAE.

Để mô tả đặc điểm phân bố dữ liệu tại không gian lớp ẩn, giúp tường minh hơn về xu hướng dịch chuyển của các nhóm tấn công khi được thực thi bởi mô hình SAE và DSAE. Luận án sử dụng kỹ thuật minh hoạ các vector này trên không gian hai chiều dựa trên khoảng cách Euclid từ các vector này đến gốc toạ độ trong không gian lớp ẩn. Khi minh hoạ trên biểu đồ hai chiều (2-D), các toạ độ x_i và y_i của mỗi vector lớp ẩn z_i được tính toán theo mô tả tại Thuật toán 2.2. Để tiện cho minh hoạ, với các nhóm tấn công có lớn hơn 500 mẫu dữ liệu, luận án lấy ngẫu nhiên 500 mẫu dữ liệu cho biểu diễn, riêng nhóm tấn công U2R lấy toàn bộ.

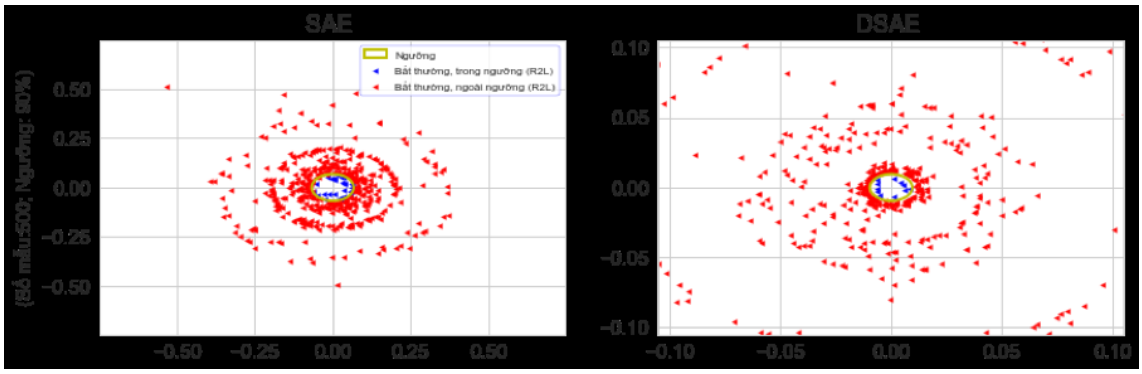
Thuật toán 2.2 Minh hoạ vector (lớp ẩn AE) trong không gian 2 chiều (2D)

INPUT: Tập vector lớp ẩn Z.
OUTPUT: Tập toạ độ trên không gian 2-D, P.

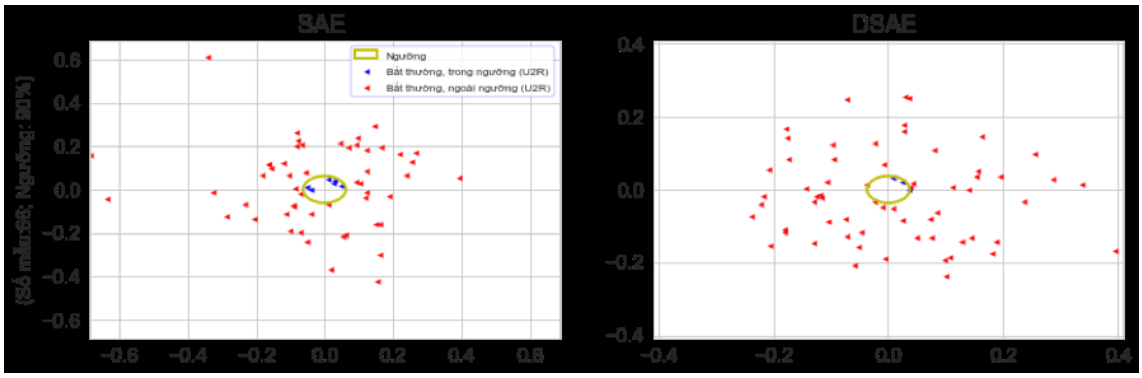
$P \leftarrow [...]$
Lấy số phần tử tập cần minh hoạ, $nu\_max \leftarrow |Z|$
$i \leftarrow 0$
while $(i < nu\_max)$ **do**
    Tính khoảng cách đến gốc toạ độ, $(dz_i) \leftarrow \|Z[i]\|$
    Giá trị ngẫu nhiên $\alpha \leftarrow$ Random [0...360]
    $x_i \leftarrow \cos\alpha * dz_i$
    $y_i \leftarrow \sin\alpha * dz_i$
    $P \leftarrow p(x_i, y_i)$
    i++
end while
Trả về P.



Hình 2.8: *Không gian lớp ẩn nhóm tấn công DoS trên SAE, DSAE*



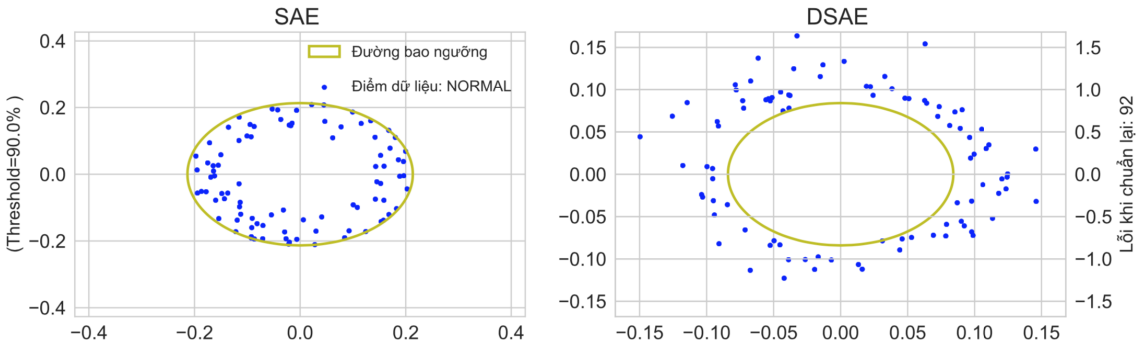
Hình 2.9: *Không gian lớp ẩn nhóm tấn công R2L trên SAE, DSAE*



Hình 2.10: *Không gian lớp ẩn nhóm tấn công U2R trên SAE, DSAE*

Số liệu minh hoạ chuyển dịch của vector lớp ẩn được thể hiện trên các Hình 2.7, 2.8, 2.9 và 2.10. Các vòng tròn màu vàng thể hiện cho ngưỡng, là ranh giới phân tách giữa bình thường và bất thường. Các điểm màu đỏ thể hiện các mẫu dữ liệu tấn công, các điểm màu xanh thể hiện là điểm tấn công khó với SAE. Quan sát các hình cho thấy, các mẫu tấn công khó với SAE có vector lớp ẩn xu hướng bị đẩy ra xa gốc toạ độ hơn khi thực thi bởi DSAE, xảy ra trên cả 04 nhóm tấn công Probe, Dos, R2L, U2R. Thêm vào đó, nhóm tấn công R2L cho phân bố vector lớp ấn khác hơn, mật độ dày đặc theo hướng gần gốc toạ độ hơn. Điều này cũng phù hợp với nhận định các tấn công R2L thường ẩn thông tin mã độc trong các nội dung gói tin, do vậy R2L có dữ liệu giống với lưu lượng mạng bình thường và làm cho R2L thường khó bị phát hiện hơn [56]. Mô phỏng này cũng phù hợp với kết quả đã đề cập ở trên là DSAE cho kết quả tốt hơn SAE trên nhóm tấn công R2L.

Để mô tả rõ hơn về hiệu quả của DSAE và SAE trên nhóm tấn công mà DSAE thể hiện thế mạnh hơn, cụ thể là R2L, các chỉ số độ chính xác được sử dụng. Bảng 2.5 trình bày giá trị False Alarm Rate (FAR) và Detection rate (DR) của SAE và DSAE trên loại tấn công R2L. Kết quả này được tính toán sử dụng ngưỡng quyết định là giá trị AS của tập huấn luyện, tại đấy 90% mẫu dữ liệu của tập huấn luyện khi tham gia kiểm thử được cho là bình thường. Từ bảng này cũng cho thấy, với loại dữ liệu R2L, DSAE thể hiện hiệu quả hơn SAE ở cả hai chỉ số DR và FAR. Cụ thể FAR của DSAE vừa thấp hơn so với SAE (0.102



Hình 2.11: *Minh hoạ các điểm bình thường đã được phân lớp đúng bởi SAE nhưng lại phân lớp sai bởi DSAE*

và 0.104) và DR của DSAE lại cao hơn khá nhiều so với SAE (0.697 và 0.655).

Tổng thể lại, kết quả trong thực nghiệm cho thấy DSAE có hiệu quả tương đồng với SAE trên các tập dữ liệu loại tấn công mạng phổ biến. Nhưng DSAE cho thấy khả năng phát hiện bất thường hiệu quả hơn phương pháp NAD tiêu biểu dựa trên học sâu, cụ thể SAE, trên loại tấn công R2L.

Mặc dù thực nghiệm cho thấy DSAE hiệu quả hơn SAE đối với các tấn công SAE gặp khó, tuy vậy cơ chế hoạt động của DSAE theo hướng cố để điều hướng vector lớp ẩn của các tấn công khó ra xa gốc toạ độ trong không gian lớp ẩn. Điều này dẫn đến một số các mẫu dữ liệu bình thường vốn dĩ đã được lần co thứ nhất thực hiện đúng, nhưng ở lần co sau đã bị SAE phân tách sai.

Khi quan sát các mẫu dữ liệu bình thường này trên tập dữ liệu R2L, kết quả như tại Hình 2.11. Trong hình, các điểm màu xanh thể hiện điểm dữ liệu bình thường tương ứng đã phân tách tốt ở SAE nhưng lại bị phân tách sai ở DSAE. Nhưng tổng thể, đối với nhóm tấn công R2L kết quả cho thấy số lượng bị phân tách sai của DSAE ít hơn khá nhiều so với số lượng bị phân lớp sai bởi SAE.

Như đã đề cập, DSAE sử dụng mặc định vector z_2 để biểu diễn dữ liệu ở không gian đầu ra, phục vụ cho việc phân tách trạng thái bình thường và bất thường. Tuy nhiên cơ chế hoạt động của DSAE dẫn đến cả z_1 và z_2 đều có những lợi thế khác nhau trong phân tách bất thường, mặc dù về lý thuyết z_2 được đánh giá hiệu quả hơn. Điều này cũng mở ra cơ hội trong tìm kiếm giải

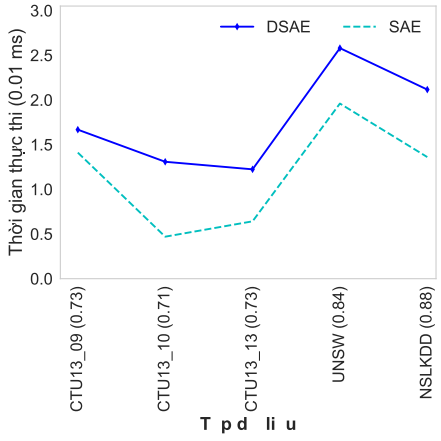
pháp kết hợp cả hai đầu ra của DSAE là z_1 và z_2 (ký hiệu hai mô hình tương ứng là DSAE_Z1 và DSAE_Z2) để tạo ra mô hình có thể phân tách được tốt hơn giữa bình thường và bất thường.

Ngoài ra, đối với mô hình DSAE, luận án đã tiến hành thực nghiệm với mô hình hoạt động như DSAE nhưng số lần co là ba và bốn lần. Việc thực nghiệm được tiến hành với các tập dữ liệu và thiết lập tham số như đã làm với DSAE. Tuy nhiên kết quả thực nghiệm cho thấy kết quả đạt được không tốt hơn so với SAE. Điều này có thể giải thích, vì bản chất của DSAE là muốn sử dụng cả giá trị lỗi tái tạo và xu hướng của vectơ lớp ẩn trong SAE, qua đó tạo ra mô hình có thể phân tách được tốt hơn đối với một số các mẫu bất thường mà SAE dễ bị nhầm lẫn với bình thường. Do vậy, trong trường hợp tăng thêm số lần co, ví dụ là ba lần co, thì xu hướng lỗi tái tạo sẽ càng lớn hơn (là sự khác biệt giữa mẫu dữ liệu đầu ra (X-out) lần cuối và mẫu dữ liệu gốc đầu vào). Với RE lớn ngay cả với mẫu dữ liệu bình thường thì nguy cơ các dữ liệu bình thường sẽ bị kéo xa hơn so với gốc toạ độ trong không gian lớp ẩn, do vậy việc phân tách mẫu bình thường và bất thường có thể dẫn đến không hiệu quả.

Với mạng nơ-ron học sâu, độ phức tạp thuật toán cho quá trình huấn luyện được cho là quan hệ tuyến tính đến rất nhiều các yếu tố như số lớp ẩn, số chiều của mỗi lớp. Ngoài ra các yếu tố quyết định đến độ phức tạp như hàm kích hoạt hay thuật toán đạo hàm lặp [14] được sử dụng cho quá trình huấn luyện. Cụ thể, với mạng nơ-ron AutoEncoder, độ phức tạp tính toán được cho là $\mathcal{O}(n^2)$ [21], trong đó n là số mẫu tập huấn luyện.

Như phát biểu bài toán luận án đã đề cập, nội dung luận án tập trung vào cải tiến khả năng phát hiện bất thường trên phương diện khả năng phát hiện bất thường. Do vậy, phạm vị luận án không đi sâu đến vấn đề độ phức tạp thuật toán huấn luyện các mô hình học máy. Đối với quá trình kiểm tra, độ phức tạp tính toán của mô hình dựa trên mạng nơ-ron học sâu được cho là thấp vì chỉ tính toán dựa trên một hàm với các tham số đã có sẵn [21], độ phức tạp chỉ phụ thuộc vào tập các trọng số (weights) của mạng, tương ứng với $\mathcal{O}(1)$.

Quá trình thực nghiệm để so sánh thời gian xử lý đối với một mẫu dữ liệu (query time), luận án thực hiện đo thời gian thực thi của SAE, DSAE theo phương pháp sau. Tạo các tập dữ liệu con bằng cách lấy ngẫu nhiên 1000 mẫu dữ liệu tương ứng của năm tập dữ liệu kiểm tra của CTU13_09,CT13_10, CTU13_13, UNSW-NB15, NSLKDD. Mục đích của việc lấy mẫu để tạo ra các bộ dữ liệu kiểm tra có số lượng mẫu như nhau, giúp cho việc đối chiếu kết quả được tường minh hơn. Thực nghiệm kiểm tra 100 lần đối với từng tập dữ liệu con này bằng mô hình SAE và DSAE với tham số thiết lập như đã mô tả ở phần trên. Tính toán thời gian truy vấn cho mỗi mẫu dữ liệu. Kết quả thời gian kiểm tra của mỗi mô hình trên năm tập dữ liệu được mô tả tại Hình 2.12.



Hình 2.12: *Thời gian truy vấn của phương pháp SAE, DSAE*

Hình 2.12 cho thấy, thời gian kiểm tra của SAE và DSAE trên các tập dữ liệu gần tương đồng nhau (với mẫu dữ liệu thuộc tập CTU13_09, thời gian xử lý của DSAE chỉ nhiều hơn SAE khoảng 10%).

2.5. Kết luận

Kết quả nghiên cứu trình bày trong chương đã giải quyết phát biểu bài toán đầu tiên của luận án, nghiên cứu phát triển phương phương pháp học sâu cho NAD thông qua khắc phục các hạn chế mà phương pháp tiêu biểu đang gặp

phải, phương pháp SAE. Hai hạn chế chính của phương pháp SAE gồm: (1) hoạt động không hiệu quả với đối tượng quan sát có trạng thái bình thường tồn tại ở nhiều cụm; (2) gặp khó khăn với một số tấn công nhất định. Trong đó, hạn chế thứ nhất nằm ở khâu xử lý dữ liệu trước SAE, hạn chế còn lại nằm ở phần lõi của chính SAE.

Để khắc phục hạn chế của SAE được cho là không hiệu quả với các tập dữ liệu hiện hữu ở dạng nhiều cụm. NCS đề xuất cải tiến thông qua kết hợp thuật toán phân cụm với SAE, phương pháp có tên KSAE (Clustering-based Shrink-AutoEncoder). KSAE chia dữ liệu huấn luyện theo K cụm, với K được phát hiện bởi các phương pháp phổ biến như Elbow, tiếp đó SAE được huấn luyện trên các cụm tương ứng. Việc tiền xử lý dữ liệu để SAE được huấn luyện với tập dữ liệu có tính đồng nhất (theo từng cụm) góp phần làm cho mô hình có khả năng phân biệt giữa mẫu dữ liệu bình thường và bất thường tốt hơn. Kết quả thực nghiệm trên các bộ dữ liệu phổ biến, hiện đại trong lĩnh vực an ninh mạng đã góp phần khẳng định cho lý thuyết. Theo đó thuật toán đề ra cải tiến khả năng làm việc của SAE khi ứng dụng trong môi trường mà dữ liệu quan sát tồn tại ở dạng nhiều cụm.

Để khắc phục hạn chế thứ hai, luận án cũng đề xuất giải pháp DSAE, phát triển mở rộng nội tại nhân SAE để có thể phân tách tốt hơn các bất thường mà SAE gặp khó. Khác với hầu hết các phương pháp học sâu dựa trên AutoEncoder khác, DSAE sử dụng cả lỗi tái tạo RE và cơ chế điều hướng lớp ẩn trung tâm của SAE để phục vụ cho mục đích phân tách các tấn công được cho là khó với SAE. Kết quả thực nghiệm cho thấy DSAE hoạt động hiệu quả hơn so với SAE trên các tấn công mà SAE gặp khó khăn. Cụ thể ở đây là nhóm tấn công R2L, đây là loại tấn công mà nội dung của nó được nhúng trong các gói tin, qua đó giúp cho dữ liệu mà nó sinh ra giống với lưu lượng mạng bình thường. Thêm vào đó, kết quả thực nghiệm cũng cho thấy, phương pháp cải tiến DSAE cho thời gian thực thi kiểm tra tuyến tính với phương pháp SAE và chi phí tính toán giữa hai phương pháp là gần tương đương nhau.

Kết quả nghiên cứu trong chương đã góp phần cải tiến khả năng phát hiện bất thường so với các phương pháp tiêu biểu NAD dựa trên học sâu hiện tại. Tuy vậy các phương pháp đơn lẻ này vẫn cần được phát triển theo hướng tổng hợp dữ liệu, để từ đó có thể khắc phục được hạn chế của một phương pháp đơn, đó là thường chỉ tốt trên một bài toán (tập dữ liệu) cụ thể mà lại không hiệu quả trên các tập dữ liệu khác. Ngoài ra, các phương pháp đơn OCC cho NAD thường yêu cầu phải có sự hỗ trợ của các chuyên gia trong việc thiết lập ngưỡng ra quyết định. Do vậy, nghiên cứu giải pháp cho NAD cũng cần phải giải quyết vấn đề này, luận án trình bày giải pháp trong chương tiếp theo.

CHƯƠNG 3. PHÁT HIỆN BẤT THƯỜNG DỰA TRÊN TỔNG HỢP DỮ LIỆU

Chương này trình bày nội dung nghiên phát triển mô hình phát hiện bất thường dựa trên tổng hợp dữ liệu. Nội dung trình bày trong chương là giải quyết hai vấn đề đã đặt ra trong phát biểu bài toán của luận án: (i) khắc phục hạn chế của phương pháp đơn, các phương pháp đơn được cho thường chỉ tốt trên một bài toán (môi trường, tập dữ liệu hay tấn công) cụ thể mà không tốt trên các bài toán khác; (ii) giải pháp NAD cần tiến tới khả năng tự ước lượng ngưỡng ra quyết định để phù hợp theo yêu cầu ứng dụng thực tế.

Nội dung của chương trình bố cục theo các phần sau. Đầu tiên, phân tích làm rõ vấn đề nội dung nghiên cứu đặt ra. Tiếp theo, đề xuất giải pháp thực hiện, phần này tập trung trình bày mô hình khung NAD dựa trên tổng hợp dữ liệu, sử dụng lý thuyết D-S để kết hợp quyết định từ các nguồn bao gồm cả các phương pháp OCC truyền thống và học sâu. Sau đó, trình bày phương pháp thực nghiệm; phân tích kết quả, đánh giá giải pháp đề xuất. Trong phần cuối của chương trình bày kết luận. Các kết quả chính trình bày được công bố trong các công trình khoa học [CT2], [CT3], [CT6] (trong phần CÁC CÔNG TRÌNH CÓ LIÊN QUAN ĐẾN LUẬN ÁN).

3.1. Giới thiệu

Như đã đề cập ở phần mở đầu, các bộ phân lớp đơn dựa trên OCC được xem là phù hợp nhất để xây dựng giải pháp phát hiện bất thường. Nguyên nhân vì các phương pháp OCC cho phép xây dựng các thuật toán phát hiện chỉ từ dữ liệu bình thường, khi việc gán nhãn cho dữ liệu bất thường mạng hết sức khó khăn, việc đảm bảo gán đủ nhãn để đại diện cho các bất thường mạng được

cho là không khả thi [20], [77]. Các thuật toán OCC truyền thống điển hình như One-class Support Vector Machine (OCSVM) [88], Local Outlier Factor (LOF) [16] và Kernel Density Estimation (KDE). Những thuật toán này đã cho kết quả rất tốt trên nhiều lĩnh vực phát hiện bất thường, tuy vậy lại gặp thách thức với các dữ liệu phức tạp, có số chiều và kích thước rất lớn [20].

Mạng nơ-ron học sâu dựa trên AutoEncoder (Deep AutoEncoder - DeAE) được xem là phương pháp tiên tiến trong lĩnh vực phát hiện bất thường, nhờ khả năng làm việc hiệu quả với dữ liệu quan sát có số chiều và kích thước rất lớn [18], [19], [20]. Một số nghiên cứu phát triển sử dụng AutoEncoder theo theo hướng giải quyết một số hạn chế mà mô hình tiêu biểu NAD dựa trên học sâu đã được trình bày trong Chương 2 của luận án. Mặc dù có các phương pháp đơn rất mạnh cho lĩnh vực phát hiện xâm nhập thì các phương pháp này vẫn nổi lên hai hạn chế chung theo sau.

Thứ nhất, vấn đề đối với phương pháp đơn phát hiện xâm nhập nói chung và phương pháp đơn phát hiện bất thường (Single AD - SglAD) nói riêng, thường được cho là hoạt động rất hiệu quả trên một số bài toán (tập dữ liệu cụ thể) mà lại không hiệu quả trên các bài toán khác [34], [57], [69], [102]. Vì như đã đề cập ở phần trước, mức độ hiệu quả của các phương pháp vẫn phụ thuộc vào dữ liệu quan sát. Trong lĩnh vực an ninh mạng, vấn đề này xuất phát từ sự thay đổi và phức tạp của hệ thống mạng, các tấn công hay xâm nhập mạng thường được tạo ra bởi con người, do vậy tính bất thường đặc biệt hơn so với các lĩnh vực khác [102], [117].

Thứ hai, làm thế nào để chọn ngưỡng phù hợp cho ra quyết định đối với mô hình NAD phát triển theo hướng OCC. Thật vậy, khi áp dụng OCC, chỉ có duy nhất dữ liệu bình thường được sử dụng cho xây dựng mô hình, mô hình sau đó được sử dụng cho kiểm tra các dữ liệu đầu vào và cho đầu ra là một đơn vị độ đo AS. Khi thiết lập một giá trị ngưỡng quyết định, với mẫu dữ liệu có AS lớn hơn ngưỡng này sẽ được xem là bất thường (Anomaly), và ngược lại là bình thường (Normal) [20], [40]. Do vậy, làm thế nào để chọn ngưỡng cho đầu ra của

mô hình vẫn là một câu hỏi cần lời giải.

Vấn đề SglAD có thể được giải quyết trên cơ sở kết hợp điểm mạnh từ các phương pháp đơn khác nhau [57], [68], [69]. Tuy vậy, làm thế nào để có thể gộp các phương pháp đơn dựa trên OCC để tạo thành một phương pháp mạnh cho phát hiện bất thường. Các kỹ thuật sử dụng nhiều phương pháp khác nhau cho xây dựng một phương pháp mạnh và hiệu quả hơn trong các bài toán phân lớp được biết đến như: lai ghép (Hybrid), học theo cộng động (Ensemble Learning) hay tổng hợp dữ liệu (Data Fusion). Trong số đó, Data Fusion được cho là phù hợp trong điều kiện bài toán đặt ra, như đã phân tích trong Chương 1. Trong phạm vi bài toán đặt ra, Data Fusion (DF) bản chất là tổng hợp quyết định từ các bộ phân lớp khác nhau. Các yếu tố cần phải được xem xét khi phát triển một hệ thống DF [10], [68], [117] gồm: (1) xác định mức hoạt động của DF, có ba mức tổng hợp là: mức dữ liệu, mức thuộc tính, và mức quyết định [68], [102], [105]. Trong đó, mức tổng hợp ở tầng quyết định thường được các nhà nghiên cứu lựa chọn nhờ tính linh hoạt và tính phù hợp cho các bài toán thực tế [68]. Bởi vì mục tiêu đặt ra là có thể tận dụng được kết quả, lợi thế từ các phương pháp đơn vốn dĩ đã rất hiệu quả; việc sử dụng DF mức quyết định cũng giúp cho mô hình có thể sử dụng các phương pháp đơn phù hợp theo yêu cầu bài toán, quá trình thay thế này không ảnh hưởng đến kết cấu hoạt động của mô hình đề xuất. Đó cũng là lý do NCS chọn mức tổng hợp này cho phát triển phương pháp phát hiện bất thường mạng. (2) cơ sở nào để chọn được các phương pháp đơn, qua đó giúp cho tận dụng được điểm mạnh của DF và nâng cao khả năng phát hiện bất thường cho hệ thống. (3) lựa chọn thuật toán tổng hợp (fusion algorithm) nào?, được cho là vấn đề cơ bản nhất khi xây dựng hệ thống DF.

Các nghiên cứu gần đây [68], [69], [82], [92], [104] cho thấy rằng, thuật toán tổng hợp dựa trên lý thuyết Dempster-Shafer (D-S) có nhiều tiềm năng cho phát triển các mô hình tổng hợp dữ liệu trong lĩnh vực an ninh mạng. Một trong các thuận lợi chính là đặc điểm linh hoạt của lý thuyết D-S, lý thuyết này có thể áp dụng mà không yêu cầu xác suất tiền nghiệm như phương pháp suy luận

nổi tiếng Bayes, do vậy D-S được xem là phù hợp cho các bài toán phát hiện thông tin chưa từng được biết đến [25]. Tuy nhiên, khi áp dụng D-S, việc định nghĩa tập giả thuyết đơn FoD (Frame of Discernment) và việc đề xuất hàm BPA (Basic Probability Assignment) thường khó và phức tạp. Thêm vào đó, để xử lý với vấn đề tổng hợp dữ liệu từ các phương pháp đơn có độ tin cậy khác nhau, việc áp dụng D-S đòi hỏi phải cải tiến hàm kết hợp DRC (Dempster-Shafer Rule Combination) của lý thuyết [69], [73], [92].

Giải pháp DF đã được áp dụng nhiều cho tổng hợp các phương pháp phân lớp đơn khác nhau nhằm tạo ra phương pháp mới có độ phân lớp chính xác hơn theo hướng học có giám sát [68], [102]. Có nghĩa là, mô hình tổng hợp dữ liệu thực hiện kết hợp quyết định từ các phương pháp học có giám sát khác nhau để đưa ra quyết định cuối cùng. Quá trình này dựa trên tham số ngưỡng quyết định của mỗi phân lớp và trọng số được gán cho từng bộ phân lớp. Điểm khác biệt ở phạm vi nghiên cứu này là các bộ phân lớp đơn đều theo hướng OCC, do vậy việc ứng dụng DF gặp rất nhiều thách thức như đã đề cập ở phần mở đầu, gồm: (i) vấn đề xác định ngưỡng quyết định cho các phương pháp đơn khi tham gia tổng hợp; (ii) làm thế nào để xác định trọng số độ tin cậy của từng phát biểu, từ từng nguồn tham gia tổng hợp. Chi tiết hơn về các vấn đề này đã được đề cập tại Phần mở đầu.

Việc ứng dụng lý thuyết D-S cho cho phép gán trọng số niềm tin cho từng dẫn chứng. Điều này giúp cho phương pháp có thể tính toán và so sánh được trạng thái hệ thống là bình thường hay bất thường dựa trên kết hợp dữ liệu thu thập được từ các bộ phân đơn lớp OCC. Theo đó có thể giúp cho giải pháp đề xuất được xây dựng trên nền tảng OCC nhưng vẫn có thể cung cấp đầu ra là nhãn nhị phân, đây được xem là yếu tố cần thiết khi triển khai mô hình NAD trong thực tế.

Theo như nghiên cứu sinh được biết, chưa có nghiên cứu đề xuất phương pháp DF từ các bộ phân lớp đơn OCC và giải quyết các vấn đề thách thức trên theo hướng sử dụng lý thuyết D-S.

Việc xây dựng được mô hình NAD theo hướng này sẽ góp phần giải quyết đồng thời hai vấn đề đặt ra: khắc phục được hạn chế của phương pháp đơn trong phát hiện bất thường; và mô hình có thể hoạt động không cần phải thiết lập ngưỡng quyết định. Giải pháp đề xuất cụ thể được trình bày trong phần tiếp theo. Các vấn đề chính mà nghiên cứu đã giải quyết trong bước này như: (1) lựa chọn các phương pháp đơn cho mô hình tổng hợp; (2) xác định ngưỡng cho các phương pháp đơn OCC; (3) đánh giá độ tin cậy của phương pháp đơn OCC; (4) xác định trọng số của phương pháp OCC khi tham gia mô hình tổng hợp; (5) áp dụng lý thuyết D-S cho mô hình khung đề xuất

3.2. Giải pháp đề xuất

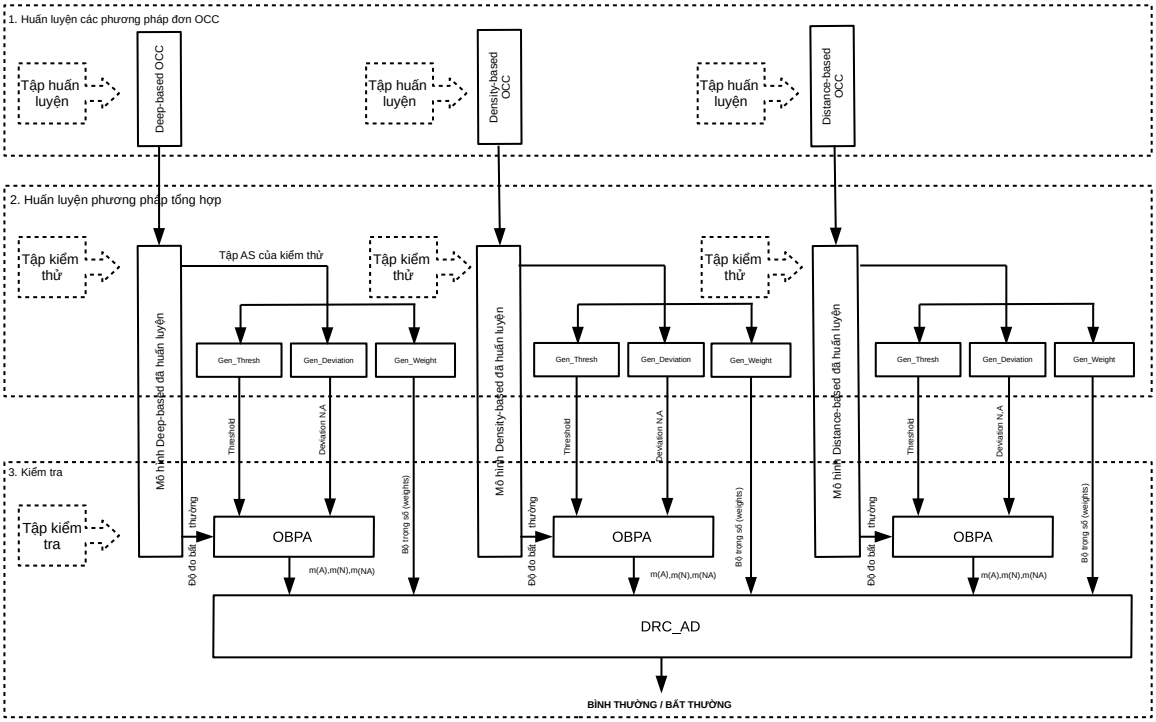
Trong phần này, luận án trình bày mô hình NAD dựa trên tổng hợp dữ liệu, giải pháp sử dụng lý thuyết D-S, để kết hợp được các quyết định từ các phương pháp đơn đã được cấu tạo theo hướng OCC, gồm cả phương pháp OCC truyền thống và OCC học sâu. Phương pháp có tên là One-class Fusion-based Anomaly Detection (OFuseAD). Hình 3.1 minh hoạ kiến trúc của OFuseAD.

3.2.1. Các thành phần của phương pháp OFuseAD

Nội dung theo sau trình bày phương án giải quyết các vấn đề chính khi xây dựng mô hình DF, nội dung trình bày tuần tự từ phương pháp đơn đến vấn đề kết hợp, theo hướng tính phức tạp tăng dần.

3.2.1.1. Lựa chọn các phương pháp đơn AD

Việc quyết định các phương pháp đơn (SglAD) nào tham gia cho mô hình DF là một vấn đề quyết định đến hiệu quả của DF. Một cách lựa chọn được khuyến nghị là trên cơ sở năng lực và mối liên hệ giữa các SglAD [69], [101]. Sử dụng thuật ngữ tương quan (correlated), không tương quan (uncorrelated) để chỉ mối liên hệ lẫn nhau giữa hai SglAD, hiệu năng (performance) để chỉ khả năng của



Hình 3.1: *Kiến trúc của giải pháp OFuseAD*

SglAD trong phân biệt bất thường và bình thường cho đối tượng đang quan sát. Khi đó với mỗi cặp SglAD, có bốn trường hợp có thể xảy ra trên khía cạnh tương quan và hiệu năng, bao gồm: (1) kết hợp hai SglAD không tương quan và hiệu năng rất khác nhau; (2) kết hợp hai SglAD rất tương quan và hiệu năng rất khác nhau; (3) kết hợp hai SglAD không tương quan và hiệu năng tương tự nhau; (4) kết hợp hai SglAD rất tương quan và hiệu năng tương tự nhau.

Thomas và cộng sự [101] chứng minh rằng, kết quả tổng hợp dữ liệu có thể đạt hiệu quả nhất khi kết hợp các nguồn thành phần, ở đó các nguồn thành phần theo từng cặp là không tương quan và có hiệu năng tương tự nhau (phương án thứ 3 như trên).

Trên cơ sở đó, trong phương pháp OFuseAD đề xuất, chúng tôi chọn các SglAD từ ba kỹ thuật phát hiện bất thường rất khác nhau. Các phương pháp đơn này gồm: dựa trên học sâu (deep learning-based); dựa trên khoảng cách (distance-based); và dựa trên mật độ (density-based). Khi áp dụng OFuseAD

vào các ứng dụng cụ thể, các phương pháp tiêu biểu theo các kỹ thuật này nên được sử dụng. Việc lựa chọn các phương pháp tiêu biểu này được giả định là đồng nghĩa với có hiệu năng tương tự như đã phân tích trên. Ví dụ khi áp dụng OFuseAD cho lĩnh vực an ninh mạng, luận án đề xuất sử dụng DSAE như đại diện cho thành phần học sâu OCC, LOF [16], KDE [111] đại diện cho thành phần OCC dựa trên khoảng cách và OCC dựa trên mật độ.

3.2.1.2. Xác định ngưỡng cho phương pháp đơn OCC

Xác định ngưỡng quyết định là rào cản chung cho các phương pháp OCC khi được triển khai trong môi trường thực tế, được xem là nhiệm vụ không hề đơn giản ngay cả với các chuyên gia [40]. Khi áp dụng OCC cho tạo mô hình phát hiện bất thường, chỉ có dữ liệu bình thường được sử dụng cho cho huấn luyện. Mô hình khi thực hiện kiểm tra sẽ trả về giá trị là AS từ các điểm dữ liệu đầu vào. Việc đưa ra ngưỡng trên độ đo bất thường sẽ giúp xác định cụ thể hơn về đối tượng đầu vào là bất thường hay không, chứ không chỉ dừng ở một giá trị xác suất [20], [40], [74].

Trong một số nghiên cứu gần đây, Cao và cộng sự [20] đưa ra mức ngưỡng ở 90% dữ liệu tập huấn luyện được phân lớp đúng. Nhóm tác giả Mirsky và cộng sự [74] xây dựng mô hình phát hiện bất thường sử dụng mạng nơ-ron AE và cho học theo phương pháp OCC, mô hình sau huấn luyện được sử dụng để kiểm tra cũng với đầu vào tập huấn luyện để thu được tập AS. Giá trị cao nhất trong tập này được chọn là ngưỡng ra quyết định. Với cách làm này, khi gặp các bộ dữ liệu huấn luyện có chứa các điểm bình thường nhưng nhiều ngoại lai, việc xác định ngưỡng như vậy sẽ không thuyết phục và cho kết quả phát hiện thấp. Nhìn chung, các phương phát đề xuất đều có các điểm mạnh, tuy vậy tính ổn định của kết quả khi xử lý với các vấn đề khác nhau của bài toán phát hiện bất thường không thực sự thuyết phục. Với một số bộ dữ liệu, ngưỡng được xác định giúp mô hình rất hiệu quả, tuy nhiên với các bộ dữ liệu khác thì cho kết quả không cao [40].

Luận án giới thiệu một phương pháp tự động ước lượng ngưỡng cho các phương pháp đơn OCC khi tham gia mô hình tổng hợp dữ liệu, chỉ sử dụng dữ liệu bình thường. Xem xét tập kiểm thử cũng chỉ toàn dữ liệu bình thường, va, khi xem xét đầu ra của va qua bộ phân lớp f_j, là bộ phân lớp thứ j trong mô hình tổng hợp. S^{va} là tập độ đo AS cho va được trả về từ f_j. Giá trị mật độ của một vùng khi ánh xạ S^{va} lên trục AS để chỉ số mẫu dữ liệu của va cho đầu ra nằm trong vùng đó. Vùng xem xét ngưỡng là vùng giá trị AS của tập va mà ở đó có mật độ thấp hơn hẳn so với vùng còn lại. Sử dụng p_{lower} và p_{upper} để chỉ giá trị phần trăm (phần trăm va được phân loại là bình thường) ứng với cận dưới và cận trên của vùng này. Cận trên, p_{upper}, được đưa ra để nhằm loại bỏ các ngoại lai có thể xuất hiện trong tập huấn luyện, gồm chỉ dữ liệu bình thường.

Thuật toán 3.1 Thiết lập ngưỡng tự động cho OCC Gen_Threshs

INPUT: Bộ phân lớp OCC f_j, tập kiểm thử va, cận dưới vùng xem xét p_{lower} và cận trên p_{upper}..
OUTPUT: $auto\_thresh$.

Kiểm tra với bộ phân lớp OCC $S^{va} \leftarrow f_i(va)$
Sắp xếp lại $S^{va} \leftarrow sort(S^{va}, ascending\ order)$
Giá trị AS ứng với cận dưới $s_{lower} \leftarrow S^{va}[p_{lower}]$
Giá trị AS ứng với cận trên $s_{upper} \leftarrow S^{va}[p_{upper}]$
Thiết lập mật độ khởi tạo $vol \leftarrow size(S^{va})$
Thiết lập số lần chạy $num\_interval \leftarrow 2$
repeat
    kích thước một vùng $interval \leftarrow (s_{upper} - s_{lower})/num\_interval$
    $k \leftarrow 0$
    for $k \leftarrow 0$ to $num\_interval - 1$ **do**
        $s_1 \leftarrow s_{lower} + interval * (k)$
        $s_2 \leftarrow s_{upper} + interval * (k + 1)$
        $vol \leftarrow size[s_1,\ s_2]$
        if $vol == 0$ **then**
            hoàn thành việc xác định ngưỡng, break
        end if
    end for
    $num\_interval++$
until $vol! = 0$
$thresh \leftarrow s_{lower} + k * interval$
Trả về $auto\_thresh$

Theo đó, ngưỡng quyết định cần tìm là giá trị tương ứng tại đó có mật độ vùng bé nhất trong "vùng xem xét ngưỡng". Theo hướng đó, chúng tôi xây dựng thuật toán xác định ngưỡng như mô tả tại Thuật toán 3.1, trên sơ đồ thiết kế OFuseAD (Hình 3.1) tương ứng là mô-đun Gen_Thresholds. Khi xét với một SglAD. Thuật toán thực hiện đệ quy chia nhỏ vùng xem xét ngưỡng thành k vùng con khác nhau, khái niệm mật độ *vol*, tương ứng với số điểm tập S^{va} nằm trong từng vùng đó. Tiến trình này chỉ dừng lại khi xuất hiện một vùng với *vol* bằng không, và ngưỡng của phương pháp SglAD này là giá trị S^{va} tại vị trí này.

3.2.1.3. Độ tin cậy của phương pháp đơn OCC

Khi tối ưu một hệ thống tổng hợp dữ liệu, như ODS trong [69], mỗi bộ phân lớp đơn trong mô hình tổng hợp được gán trọng số niềm tin (hay gọi là khả năng khái quát hoá generalization ability), chỉ khả năng của bộ phân lớp về mức độ hiệu qua trong phân tách các lớp dữ liệu. Trong học có giám sát, niềm tin này được ước lượng sử dụng tập kiểm thử chứa nhãn của tất cả các lớp. Tuy nhiên với bộ phân lớp OCC, tập kiểm thử không sẵn có nhãn của các lớp được phân tách, do vậy thiếu cơ sở để xác định trọng số niềm tin này.

Để hiệu quả sử dụng lý thuyết D-S cho tổng hợp các OCC, luận án đề xuất một giải pháp cho ước lượng trọng số niềm tin này từ tập kiểm thử chứa chỉ dữ liệu bình thường. Chỉ số này tạm gọi là mức độ sinh lỗi (generalization error, ký hiệu *gen_error*), được định nghĩa như tại Công thức 3.1. Ý tưởng là, với một OCC f_j, chúng ta chọn ngưỡng t^i để f_j có thể phân lớp đúng tập huấn luyện với TN^i_{train}. Với $gen\_error^i_j$ là mức độ sinh lỗi của f_j tại ngưỡng t_i, giá trị này chỉ sự khác nhau giữa TN^i_{train} và TN^i_{va} tại ngưỡng t^i. Trong thực nghiệm, chúng tôi chọn mười tám ngưỡng ứng với TN_{train} từ $[90.0\%, 98.5\%]$, cách nhau 0.5%. Độ lệch tuyệt đối tại các ngưỡng này của bộ phân lớp j được định nghĩa là mức độ sinh lỗi như Công thức 3.1.

$$gen\_error_j = \frac{1}{k}\sum_{i=1}^{k}\left|TN^i_{train} - TN^i_{va}\right| \tag{3.1}$$

Với k là số ngưỡng được lấy, t_i là ngưỡng thứ i, và TN^i_{train}, TN^i_{va} là giá trị TN của bộ phân lớp j trên tập huấn luyện và tập kiểm thử tương ứng. TN^i_{va} được tính theo Công thức 3.2.

$$TN^i_{va} = \frac{\sum(f_j(va) \leqslant t_i)}{|va|} \tag{3.2}$$

3.2.1.4. Trọng số phương pháp đơn khi tham gia tổng hợp

Các phương pháp đơn khi tham gia tổng hợp có mối liên hệ chặt chẽ với nhau vì chúng cùng được huấn luyện và quan sát cùng một đối tượng dữ liệu. Để nâng cao hiệu quả trong ứng dụng lý thuyết D-S cho mô hình tổng hợp từ các phương pháp đơn OCC, vấn đề tự động xác định trọng số của các phương pháp đơn OCC tham gia tổng hợp cần được xem xét. Chúng tôi đề xuất phương án tính trọng số này trên cơ sở độ đo sinh lỗi (generalization error), theo đó OCC với độ đo này bé nên có đóng góp lớn hơn cho mô hình tổng hợp ra quyết định. Theo NCS được biết, chưa có nghiên cứu thực hiện tối ưu mô hình tổng hợp nào đề xuất phương án đánh trọng số tự động cho các OCC tham gia.

Trọng số của OCC thứ j có thể được tính toán theo Công thức 3.3.

$$w_j = \frac{min\,[gen\_error_1, gen\_error_2.., gen\_error_m])}{gen\_error_j} \tag{3.3}$$

trong đó m là số lượng các OCC tham gia mô hình tổng hợp. Giá trị trọng số này càng lớn, đóng góp của OCC tương ứng vào quá trình tổng hợp càng lớn.

Trong mô hình tổng quan, khối Gen_Weights đóng vai trò xác định trọng số cho các SglAD. Các giá trị trọng số thu được sẽ đóng vai trò là tham số đầu vào cho hàm kết hợp của lý thuyết D-S, khối DRC_AD như trên Hình 3.1.

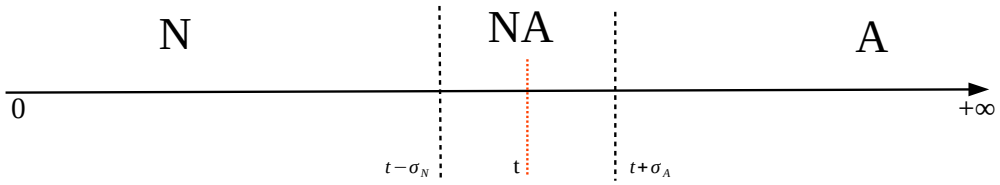
3.2.1.5. Ứng dụng lý thuyết Dempster-Shafer

Phần này trình bày thiết kế các thành phần liên quan lý thuyết Dempster-Shafer (D-S) trong kiến trúc OFuseAD cho phát hiện bất thường. Đầu tiên là định nghĩa FoD, từ thực tế lĩnh vực phát hiện bất thường mạng, giải pháp phát hiện phải chỉ ra hai trạng thái của hệ thống bình thường hay bất thường, do vậy ký hiệu N để chỉ trạng thái bình thường của hệ thống; và A để chỉ trạng thái bất thường của hệ thống. Hàm FoD theo Công thức 1.12 có thể được định nghĩa, $\Theta = \{A, N\}$, và tập giả thuyết đầy đủ $P(\Theta) = (A, N, NA, \emptyset)$, trong đó $N \cap A = \emptyset$.

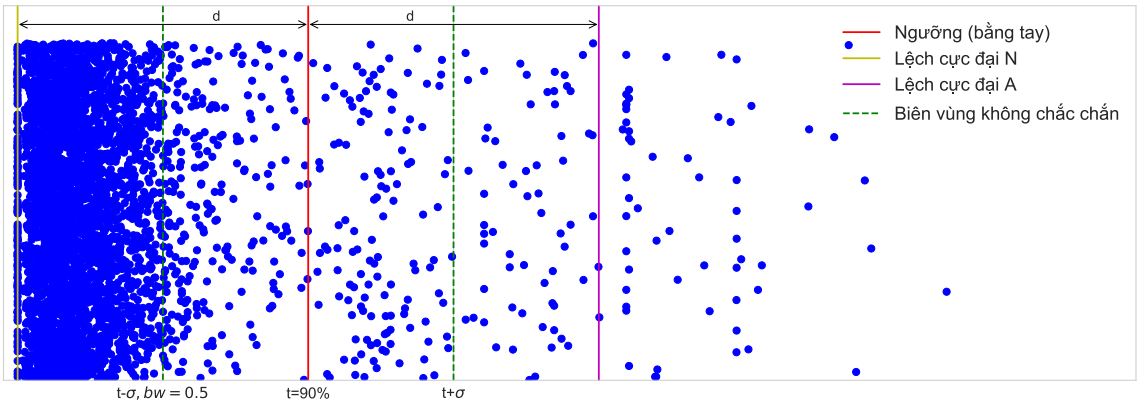
Vấn đề xây dựng hàm BPA theo lý thuyết D-S là bước phức tạp hàng đầu và cũng là yếu tố giúp cho lý thuyết D-S có thể thể áp dụng phù hợp cho các bài toán khác nhau. Hàm BPA đề xuất phải tiên quyết đáp ứng điều kiện Công thức 1.13, cụ thể ở đây là $m(A) + m(NA) + m(N) = 1$. Bản chất của việc xây dựng hàm BPA là tìm ra được tương quan giữa giữa đầu ra của một truy vấn và trục độ đo bất thường (Anomaly Score - AS) của bộ phân lớp tương ứng.

Để tìm mối liên hệ này trong điều kiện bài toán OCC, nghĩa là chỉ có dữ liệu bình thường cho xây dựng mô hình. Một số khái niệm liên quan đến ý tưởng xây dựng BPA như sau: (i) Ngưỡng quyết định (Decision Threshold - t), là giá trị của AS ứng với bộ phân lớp xem xét, tại đó t% tập dữ liệu huấn luyện (tập dữ liệu bình thường) được cho là phân lớp đúng. (ii) Tham số ,t, giúp chia tập AS thành ba vùng giá trị theo như Hình 3.2: vùng bình thường (N), vùng bất thường (A) và vùng không xác định (uncertain area -N/A). Vậy cơ sở nào xác định các giá trị σ_N, σ_A theo điều kiện bài toán OCC.

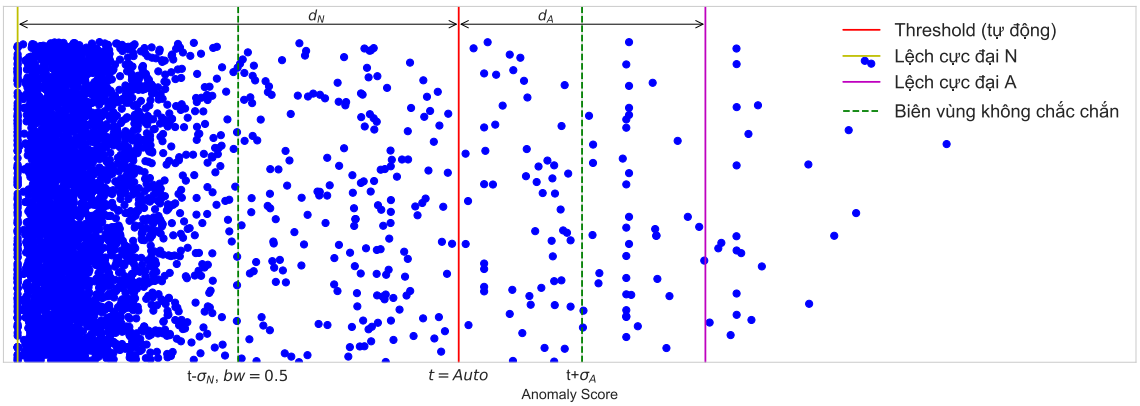
Khi quan sát tập các AS của tập dữ liệu bình thường, minh hoạ kết quả trên không gian 2D như tại Hình 3.3 và 3.4, nhận thấy. Ngưỡng t có thể xác định bằng tay (từ sự hỗ trợ của các chuyên gia) hoặc tự động xác định (như đã trình bày). Các biên σ_N, σ_A để thiết lập ba vùng cho trục AS có thể được xác định thông qua các đại lượng độ lệch cực đại và một hệ số điều chỉnh toàn cục, bw. Ở trường hợp đơn giản, có thể xác định $\sigma = \sigma_N = \sigma_A$ thông qua lựa chọn độ



Hình 3.2: *Ba vùng trên trục độ đo bất thường N, A và NA*



Hình 3.3: *Minh hoạ việc phân tách ba vùng N, A, NA theo phương án 1.*



Hình 3.4: *Minh hoạ việc phân tách ba vùng N, A, NA theo phương án 2.*

lệch cực đại như nhau cho mỗi vùng N và A ($d = d_N = d_A$, thông qua $\sigma = bw * d$ và d được xác định là khoảng cách từ ngưỡng t đến điểm cho AS bé nhất. Ở trường hợp phức tạp hơn và phù hợp với đặc tính tự nhiên của dữ liệu hơn, vì giới hạn vùng N và A thường không như nhau. Lúc này, giá trị biên phía N, σ_N được thiết lập bằng $d_N * bw$ và phía A, σ_A được thiết lập bằng $d_A * bw$. Với d_N là khoảng cách từ ngưỡng xem xét đến điểm có AS bé nhất, còn d_A là khoảng cách từ ngưỡng tới mẫu dữ liệu có AS lớn nhất trong điều kiện giả định tập dữ liệu cho huấn luyện không có các ngoại lai.

Bản chất của hàm BPA là xác định giá trị trọng số niềm tin $m(A), m(N), m(NA)$ ứng với đầu ra của một truy vấn, là độ đo bất thường AS của các bộ phân lớp đơn. Các giá trị này được xác định trên tương quan của điểm dữ liệu đầu ra trên trục AS với ba vùng đã xác định N, A, NA. Do vậy, trong điều kiện áp dụng lý thuyết D-S cho bài toán OCC, luận án đề xuất hàm OBPA (One-Class Basic Probability Assignment) như được mô tả tại Thuật toán 3.2.

Thuật toán 3.2 Thiết lập tham số cơ sở OBPA

INPUT: Độ đo AS s_i, ngưỡng t, trọng số cơ sở ($b0$, $p0$, $u0$), độ lệch cực đại (d_N, d_A), hệ số điều chỉnh bw.
OUTPUT: mA, mN, mNA.

Tính giá trị biên phía N $\sigma_N \leftarrow d_N * bw$.
Tính giá trị biên phía A $\sigma_A \leftarrow d_A * bw$.
if $s_i \leqslant t$ **then**
    Tính hệ số ứng với 'Vùng N' $b1 \leftarrow (t - s_i)/\sigma_N$, with $b1 \leq 2$.
    Tính trọng số ứng với trạng thái N, $mN \leftarrow b1 * b0$.
    Trọng số ứng với trạng thái A, $mA \leftarrow u0$.
    Trọng số ứng với trạng thái NA, $mNA \leftarrow (1 - mN - mA)$.
else
    Tính hệ số ứng với 'Vùng A', $b1 \leftarrow s_i/(\sigma_A - t)$, with $b1 \leq 2$.
    Tính trọng số ứng với trạng thái A, $mA \leftarrow bb * b0$.
    Tính trọng số ứng với trạng thái N, $mN \leftarrow u0$.
    Tính trọng số ứng với trạng thái NA, $mNA \leftarrow (1 - mN - mA)$.
end if
Trả về mA, mN, mNA

Trong đó, s_i chỉ giá trị AS cho điểm dữ liệu đầu vào x_i do SglAD tương ứng

tạo ra. Thuật toán sử dụng các khái niệm như, tham số cơ bản của niềm tin (belief) $b0$, độ hợp lý (plausibility) $p0$, và không tin (unbelief) $u0$, các tham số cơ bản này ứng với trạng thái của hệ thống là N, A hay N/A, được mô tả theo các vùng trạng thái như trên Hình 3.2.

Các giá trị này được khởi tạo rất tự nhiên $b0 \approx p0 \approx 0.5$ và một giá trị rất nhỏ $u0 = 10^{-5}$. Điều này để giúp cho các trọng số $m(A), m(NA), m(N)$ được tính toán và thoả mãn điều kiện của lý thuyết D-S, $m(A) + m(NA) + m(N) = 1$ và $m(\varnothing) = 0$ như Công thức 1.13.

Khi áp dụng hàm kết hợp DRC truyền thống của lý thuyết D-S như Công thức 1.15, tất cả phương pháp đơn AD nên có cùng vai trò, nhưng thực tế các phương pháp phát hiện bất thường thường có năng lực khác nhau [69], [70]. Để giải quyết vấn đề trên, chúng tôi sửa đổi DRC bằng cách thêm trọng số niềm tin tương ứng cho mỗi phát biểu của từng phương pháp đơn khi tham gia kết hợp. Đây có thể xem là một đóng góp quan trọng trong nghiên cứu. Cụ thể, bổ sung trọng số w_N và w_A, các trọng số này ảnh hưởng không chỉ với độ tin cậy của AD mà cụ thể đến từng giả thuyết mà các phương pháp này đưa ra đánh giá. Sự cải tiến này hoàn toàn khác với một số nghiên cứu trước như [69], các nghiên cứu này chỉ thêm trọng số cho từng AD tham gia vào hàm kết hợp DRC. Trong giải pháp đề xuất, chức năng này được thực hiện bởi khối DRC_AD, hoạt động theo như Thuật toán 3.3.

Trong thuật toán này , l_w^A và l_w^N được sử dụng để gán trọng số cho các phát biểu về trạng thái A và N tương ứng. Các giá trị trọng số này được tính như [69]. Thuật toán 3.3 cụ thể hoá công thức kết hợp nguyên bản của D-S 1.15 sau khi được cải tiến và đưa vào ứng dụng trong giải pháp đề xuất. Dựa trên giá trị đầu ra của DRC_AD gồm cả hai độ đo bất thường và bình thường tương ứng với mẫu dữ liệu quan sát. Giải pháp quyết định trạng thái của hệ thống là bất thường (Anomaly) hay bình thường (Normal) dựa trên so sánh giá trị của hai độ đo trên.

Cụ thể hơn, trong trong thiết kế OFuseAD, chúng tôi thiết lập hai thành

Thuật toán 3.3 Giải thuật DRC_AD cho phát hiện bất thường

INPUT: Các danh sách trọng số niềm tin của tập các ADs tham gia l_mA, l_mN, l_mNA, các danh sách trọng số ứng với A, N l_w^A, l_w^N.
OUTPUT: $m(A), m(N), m(NA)$.

Khai báo một danh sách rỗng $l\_mass\_ADs \leftarrow [...]$
Khởi tạo $K \leftarrow$ số phương pháp đơn ADs
Khởi tạo $i \leftarrow 0$
while $i < K$ **do**
    Tính lại trọng số niềm tin ứng với trạng thái:
    Cho trạng thái N $l\_mN[i] \leftarrow l\_mN[i] * l\_w^N[i]$
    Cho trạng thái A $l\_mA[i] \leftarrow l\_mA[i] * l\_w^A[i]$
    Cho trạng thái NA $l\_mNA[i] \leftarrow 1 - (l\_mN[i] + l\_mA[i])$
    $l\_mass\_ADs \leftarrow l\_mN[i], l\_mA[i], l\_mN[i]$
    $i++$
end while
Trọng số niềm tin kết hợp $mass(\Theta) \leftarrow drc\_combine(l\_mass\_ADs)$ 1.15
$m(A), m(N), m(NA) \leftarrow mass(\Theta)([\text{"}A\text{"}], [\text{"}N\text{"}], [\text{"}NA\text{"}])$
Trả về $m(A), m(N), m(NA)$.

phần là gen_Deviations và OBPA để thực hiện mục đích này.

Khối gen_Deviations với chức năng tạo giá trị độ lệch cực đại để cung cấp đầu vào cho khối OBPA. Khối gen_Deviations hoạt động như sau: độ lệch cực đại cho vùng xem xét phía bình thường (d_N) và phía bất thường (d_A), đầu ra tương ứng của tập kiểm thử cho SglAD đang xét là S^{va}; khi ánh xạ S^{va} vào không gian hai chiều như Hình 3.4. gen_Deviations tính toán d_A là khoảng cách từ t đến giá trị lớn nhất của tập S^{va}, còn d_N ứng với giá trị bé nhất của S^{va}.

Như đã đề cập, bw là hệ số điều chỉnh toàn cục, chúng tôi đưa ra tham số bw này với mong muốn, đây là hệ số điều chỉnh, đại diện cho tất cả các tham số khác của hệ thống và cho mô hình tổng hợp hoạt động hiệu quả ở giá trị mặc định của bw. Theo định nghĩa tại thuật toán 3.2, ta có $b1 \leq 2$ và $b1 = (t - s_i)/(d_N * bw)$, trường hợp s_i tiến về 0, d_N có giá trị tương đương ngưỡng t, dẫn đến $0.5 \leq bw$. Do vậy có thể thấy để mô hình hiệu quả, bw nên có dải hoạt động trong [0.5,1].

3.2.2. Cơ chế hoạt động của OFuseAD

Phần này mô tả cơ chế hoạt động của OFuseAD, về cơ bản, OFuseAD hoạt động trên cơ chế thu thập các thông tin có được từ các SglAD để suy luận trạng thái của hệ thống. Với mô hình DF hoạt động ở lớp tổng hợp quyết định cho phát hiện bất thường, vai trò của các phương pháp đơn không chỉ là thu thập dữ liệu mà còn cung cấp các tri thức cục bộ (hay gọi là quyết định thể hiện ở dạng xác suất cho trạng thái hệ thống). Các giá trị này sẽ là cơ sở cho việt thiết lập các thành phần của mô hình DF cũng như lý thuyết D-S. Cơ chế hoạt động của OFuseAD có thể được mô tả theo ba công đoạn: (1) Huấn luyện các SglAD; (2) Huấn luyện phương pháp tổng hợp; (3) Quá trình kiểm tra.

Trong công đoạn đầu tiên, cũng như các phương pháp dựa trên OCC khác, chỉ có dữ liệu bình thường được sử dụng cho huấn luyện các SglAD, việc huấn luyện hoàn toàn độc lập nhưng cùng một bộ dữ liệu. Kết quả huấn luyện thu được các mô hình SglAD tương ứng. Trong công đoạn thứ hai, được xem là huấn luyện cho mô hình tổng hợp, quá trình này bao gồm thực thi các phương pháp tính ngưỡng (t), độ lệch lớn nhất (d_N, d_A), trọng số tổng hợp w gán cho từng SglAD. Tập kiểm thử (validation set) được chọn cũng chỉ toàn dữ liệu bình thường nhưng độc lập với tập dữ liệu huấn luyện. Tập độ đo AS cho tập kiểm thử, S^{va} được tạo ra bởi mỗi SglAD, được sử dụng là đầu vào để huấn luyện cho DF từ các SglAD đã được huấn luyện bởi tập huấn luyện. Các khối thành phần, bao gồm gen_Thresholds, Gen_Deviations, Gen_Weights với cơ chế hoạt động đã được mô tả ở phần trên sẽ được thực thi ở công đoạn này.

Trong công đoạn kiểm thử, mỗi điểm dữ liệu đầu vào x_i của tập $X = \{x_1, x_2, ..., x_n\}$, trong R^d sẽ được đưa vào các SglAD của mô hình tổng hợp. Giá trị trả về tương ứng của mỗi phương pháp đơn, s_i đóng vai trò là đầu vào cho thành phần OBPA. Đầu ra của mỗi OBPA tương ứng với SglAD là giá trị trọng số ứng với niềm tin từng trạng thái của hệ thống do chính phương pháp đơn tương ứng đưa ra.

Cuối cùng, khối The DRC_AD thực hiện vai trò kết hợp toàn bộ các niềm tin ứng với từng trạng thái của Θ. Quyết định cuối cùng của hệ thống được đưa ra trên nguyên tắc: Nếu trọng số cho trạng thái N, $m(N)$, lớn hơn $m(A)$ ứng với trạng thái A, thì hệ thống ở trạng thái bình thường, và ngược lại thì hệ thống ở trạng thái bất thường.

3.3. Thực nghiệm

3.3.1. Dữ liệu thực nghiệm

Quá trình thực nghiệm cho đánh giá các kết quả nghiên cứu lý thuyết, chúng tôi chọn các bộ dữ liệu phổ biến và hiện đại trong lĩnh vực an ninh mạng, các bộ dữ liệu này đã được giới thiệu tại phần 1.4.1. Thông tin chi tiết về dữ liệu cho thực nghiệm được mô tả tại Bảng 3.1.

Bảng 3.1: *Các bộ dữ liệu sử dụng cho thực nghiệm*

Bộ Dữ liệu	**Số chiều nguyên bản/ sau one-hot encoding**	**Tập huấn luyện**	**Tập kiểm thử**	**Tập kiểm tra**	
				Bình thường	***Bất thường***
NSLKDD	44/122	4713	2021	9711	12833
UNSW-NB15	47/196	3920	1680	37000	45332
CTU13_08	16/40	20389	8739	43694	3677
CTU13_09	16/41	8390	3596	17981	110998
CTU13_10	16/38	4436	1902	9509	63812
CTU13_13	16/40	8942	3833	19164	24002
BoT-IoT(IoT)	10/10	258	111	107	733598
Spambase	57/57	1561	669	558	363
InternetAds	1558/1558	1107	475	396	77
SCADA	6/6	35414	15178	455329	12375

3.3.2. Thiết lập tham số thực nghiệm

Luận án xây dựng mô hình phát hiện bất thường mạng dựa trên OFuseAD như sau. Ba phương pháp SglAD được chọn là các kỹ thuật tiêu biểu trong lĩnh vực phát hiện bất thường mạng. Với SglAD mạng nơ-ron học sâu, sử dụng DSAE

(Double-Shrink AutoEncoder), gồm cả hai vector lớp ẩn là z^1 và z^2 (ứng với ký hiệu DSAE_Z1, DSAE_Z2); khoảng cách từ các vector này đến gốc toạ độ tương ứng được sử dụng trực tiếp như là AS; về hai SglAD còn lại chọn: phương pháp phát hiện dựa trên khoảng cách LOF (Local Outliner Factor) [16]; và phương pháp phát hiện dựa trên mật độ KDE (Kernel Density Estimation) [111]. Về tham số mạng nơ-ron học sâu, Kiến trúc mạng nơ-ron học sâu DSAE được thiết lập tương tự như [20]. Trọng số được khởi tạo theo phương pháp Xavier [46] và sử dụng hàm kích hoạt *tanh* cho tất cả các lớp. Chúng tôi huấn luyện mô hình với 1000 chu kỳ (epochs) sử dụng thuật toán tối ưu lặp ADADELTA. Với các phương pháp AD truyền thống: KDE sử dụng nhân Gaussian, tham số bandwidth γ được thiết lập mặt định, $\gamma = 0.1$ [86]; số lượng láng giềng gần nhất cho LOF, tương ứng với 1% kích thước tập huấn luyện [86]; sử dụng các giá trị p_{lower} là 90%, và p_{upper} là 98%. Chúng tôi thực nghiệm thí nghiệm với 10 giá trị của bw theo bước tiến 0.1, bắt đầu từ $bw_1 = 0.1$. Sử dụng kết quả $bw_5 = 0.5$ cho so sánh đánh giá.

3.4. Kết quả và đánh giá

Mô hình phát hiện bất thường được xây dựng dựa trên mô hình khung OFuseAD được thực nghiệm trên 10 tập dữ liệu phổ biến. Các tập dữ liệu này được tạo ra theo mục tiêu, vấn đề riêng của an ninh mạng. Để đánh giá mô hình, chúng tôi thực hiện các lần thực nghiệm với hai phiên bản, phiên bản thứ nhất ký hiệu là OFuseAD(ORG), mô hình NAD trong trường hợp này sử dụng hàm kết hợp DRC (Dempster-Shafer Rule Combination) nguyên bản của lý thuyết D-S. Phiên bản thứ hai sử dụng hàm kết hợp DRC_AD, là mở rộng của DRC theo như luận án đề xuất. Chỉ số so sánh khả năng phát hiện chủ yếu dựa trên F1-score như đã phân tích tại Chương 1. Mô hình tốt hơn sẽ cho F1-score tốt hơn; ngoài ra các chỉ số khác như Accuracy, DR, AUC, ROC cũng được sử dụng để phân tích thêm về kết quả.

Bảng 3.2: *Kết quả AUC của các phương pháp trên mười tập dữ liệu*

Phương pháp	Tập dữ liệu									
	SCADA (0.0)	**IoT** (0.38)	**CTU13_10** (0.71)	**CTU13_08** (0.73)	**CTU13_09** (0.73)	**CTU13_13** (0.73)	**Spambase** (0.81)	**UNSW** (0.84)	**NSLKDD** (0.88)	**InternetAds** (0.99)
DSAE_Z1	0.991	0.920	0.994	0.985	0.942	0.966	0.840	0.882	0.966	0.958
DSAE_Z2	0.991	0.956	0.992	0.986	0.931	0.972	0.827	0.903	0.963	0.959
LOF	0.993	0.967	0.999	0.982	0.973	0.988	0.743	0.893	0.850	0.831
KDE	0.991	0.999	0.999	0.985	0.808	0.944	0.818	0.888	0.938	0.927
OFuseAD(ORG) bw=0.5	0.993	0.999	0.995	0.985	0.940	0.971	0.828	0.895	0.962	0.944
OFuseAD bw=0.5	0.991	0.999	0.996	0.991	0.949	0.980	0.831	0.906	0.965	0.946

Bảng 3.3: *Kết quả F1-score của các phương pháp trên mười tập dữ liệu*

Phương pháp	Tập dữ liệu									
	SCADA (0.0)	**IoT** (0.38)	**CTU13_10** (0.71)	**CTU13_08** (0.73)	**CTU13_09** (0.73)	**CTU13_13** (0.73)	**Spambase** (0.81)	**UNSW** (0.84)	**NSLKDD** (0.88)	**InternetAds** (0.99)
DSAE_Z1	0.367	0.843	0.994	0.733	0.901	0.937	0.498	0.844	0.889	0.575
DSAE_Z2	0.489	0.995	0.995	0.708	0.885	0.927	0.420	0.848	0.906	0.555
LOF	0.449	0.991	0.995	0.755	**0.952**	0.944	0.01	0.839	0.635	0.548
KDE	0.379	0.998	0.995	0.753	0.674	0.885	0.686	0.836	0.922	0.353
OFuseAD(ORG) bw=0.5	0.490	0.995	0.996	0.758	0.930	0.945	0.687	0.845	0.918	0.598
OFuseAD bw=0.5	**0.544**	**0.999**	**0.997**	**0.770**	0.941	**0.951**	**0.690**	**0.849**	**0.925**	**0.660**

Bảng 3.4: *Kết quả ACC của các phương pháp trên mười tập dữ liệu*

Phương pháp	Tập dữ liệu									
	SCADA (0.0)	**IoT** (0.38)	**CTU13_10** (0.71)	**CTU13_08** (0.73)	**CTU13_09** (0.73)	**CTU13_13** (0.73)	**Spambase** (0.81)	**UNSW** (0.84)	**NSLKDD** (0.88)	**InternetAds** (0.99)
DSAE_Z1	0.909	0.729	0.989	0.954	0.843	0.931	0.718	0.837	0.882	0.772
DSAE_Z2	0.945	0.989	0.991	0.947	0.821	0.921	0.694	0.845	0.898	0.753
LOF	0.935	0.982	0.991	0.958	**0.921**	0.938	0.585	0.837	0.677	0.850
KDE	0.913	0.997	0.991	0.957	0.571	0.879	0.735	0.825	0.906	0.404
OFuseAD(ORG) bw=0.5	0.942	0.989	0.991	0.957	0.901	0.930	0.725	0.838	0.903	0.824
OFuseAD bw=0.5	**0.956**	**0.997**	**0.994**	**0.961**	0.904	**0.946**	**0.740**	**0.846**	**0.913**	**0.847**

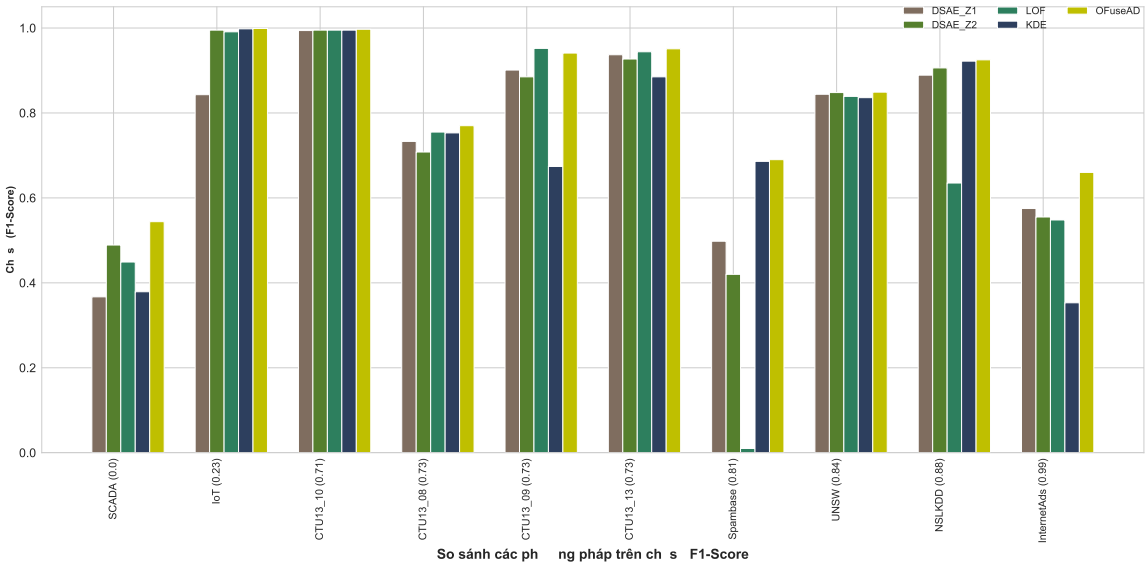
Có thể thấy từ Bảng 3.2, 3.3 và 3.4, các phương pháp đơn OCC có hiệu năng khá khác nhau trên cùng một bài toán (tập dữ liệu). Mô hình tổng hợp theo đề xuất khi sử dụng DRC truyền thống, OFuseAD(ORG), thể hiện khá tốt khả năng so với các phương pháp đơn AD (gồm: DSAE_Z1, DSAE_Z2, LOF và KDE). Trên chỉ số đo F1-score, OFuseAD(ORG) cho giá trị hiệu quả hơn với tất cả các OCC đơn trên đa số (6/10) tập dữ liệu, gồm: SCADA, CTU13_10, CTU13_8, CTU13_13, Spambase, InternetADs. Và giá trị F1-score này lớn hơn trung bình của tất cả các phương pháp đơn OCC trên cả 10 tập dữ liệu.

Còn khi so sánh OFuseAD(ORG) và mô hình đề xuất khi sử dụng giải pháp hàm DRC_AD được luận án đề xuất, số liệu từ các bảng 3.2, 3.3 và 3.4 cũng cho thấy OFuseAD cho hiệu quả hơn OFuseAD(ORG) trong hầu hết các chỉ số và các tập dữ liệu. Điều này cho thấy hàm DRC_AD đề xuất có lợi thế nhất định với hàm DRC nguyên bản trong điều kiện bài toán đặt ra.

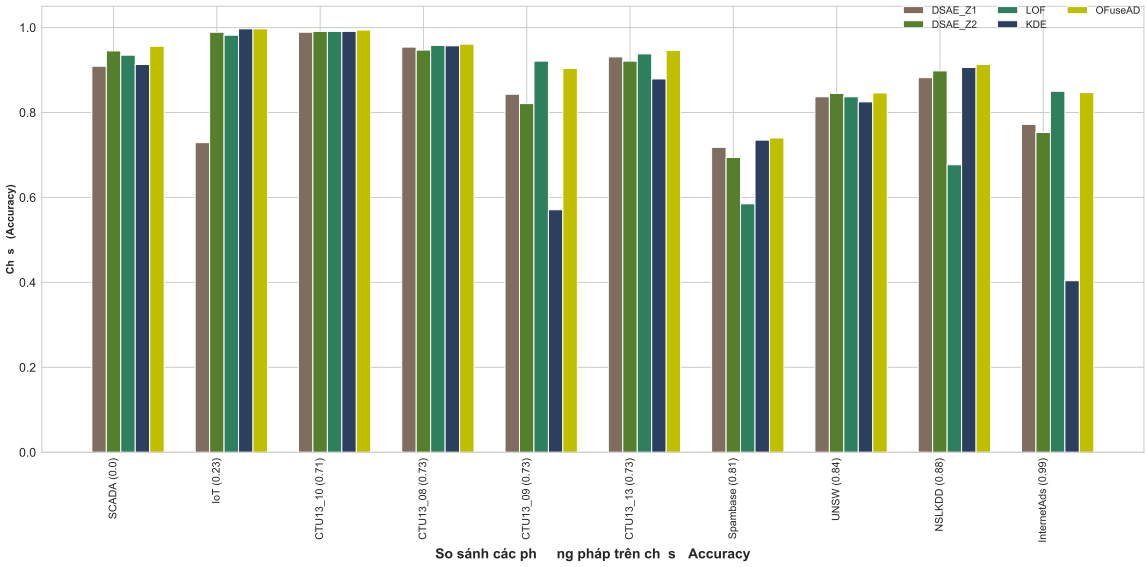
Khi so sánh mô hình đề xuất OFuseAD với các phương pháp đơn, một số đánh giá kết quả như sau. Từ kết quả Bảng 3.3, 3.4 cho thấy mô hình OFuseAD cho khả năng (theo các đơn vị F1-Score và ACC) tốt hơn các phương pháp đơn trên hầu hết tập dữ liệu (9 trên 10, ngoại trừ tập dữ liệu CTU13_09).

Kết quả này cũng được thể hiện trên Hình 3.5, 3.6; các sơ đồ này thể hiện các phương pháp đơn có khả năng phát hiện khác nhau trên cùng một bài toán, nhưng ngược lại, OFuseAD cho kết quả ổn định và khả năng phát hiện bất thường hiệu quả. Điều này chỉ ra rằng, phương pháp khung OFuseAD hoạt động khả thi và hiệu quả cho bài toán phát hiện bất thường mạng, mô hình NAD được xây dựng từ OFuseAD có thể tổng hợp được tri thức từ các nguồn, gồm cả nguồn từ kỹ thuật học sâu và kỹ thuật học máy truyền thống. Thêm vào đó, kết quả cũng chỉ ra rằng mô hình dựa trên OFuseAD cho bài toán an ninh mạng có khả năng khái quát hoá (generalization ability), vì có thể giải quyết tốt với rất nhiều các bài toán khác nhau, trong khi đó các SglAD thường chỉ tốt trên một số bài toán (môi trường) cụ thể.

Tính hiệu quả của giải pháp đề xuất có thể giải thích như sau. Trên cùng một



Hình 3.5: *Biểu đồ so sánh F1-score giữa các phương pháp trên mười tập dữ liệu*



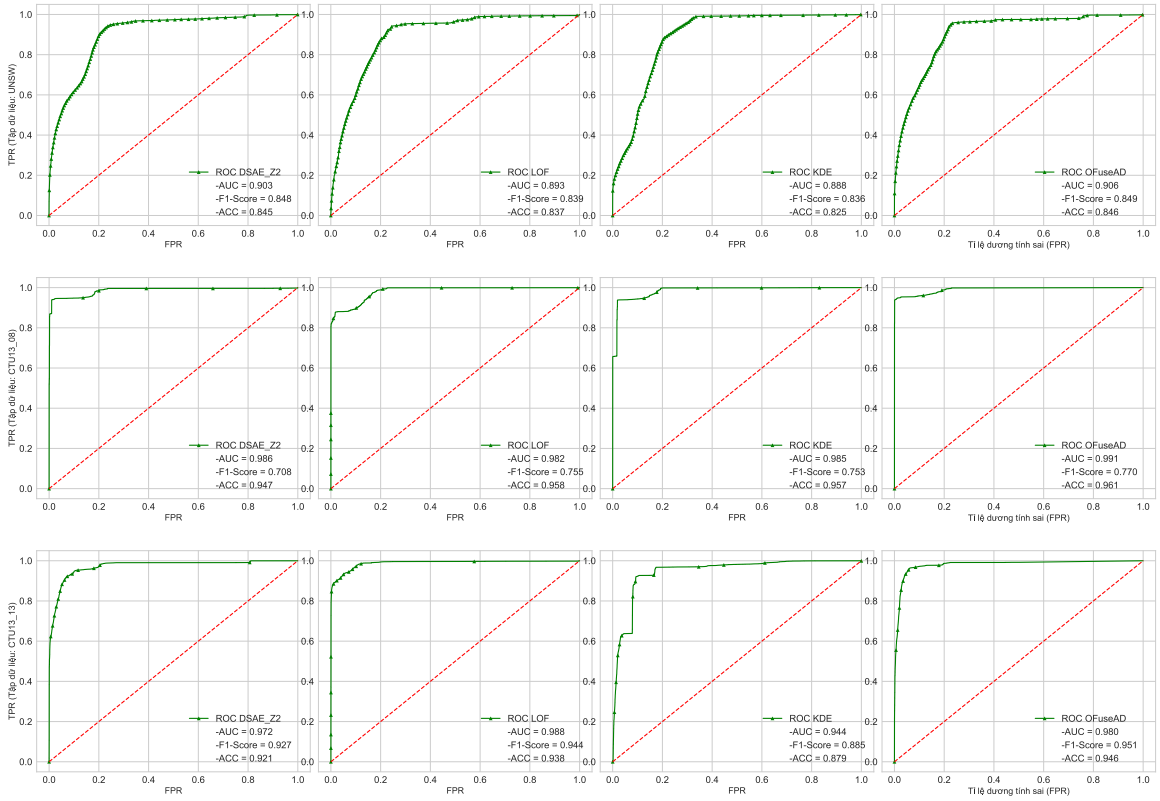
Hình 3.6: *Biểu đồ so sánh ACC giữa các phương pháp trên mười tập dữ liệu*

tập dữ liệu, một phương pháp đơn sẽ cho kết quả phát hiện bất thường cao tại một ngưỡng quyết định của nó t^1; tuy nhiên các SglAD khác sẽ cho kết quả tốt trên ngưỡng khác, là t^2. Giải pháp OFuseAD thực hiện tổng hợp các quyết định từ tất cả các quyết định cục bộ, tại các ngưỡng cục bộ khác nhau. Do vậy mô hình OFuseAD thường cho kết quả F1-score và độ chính xác phân lớp ACC tốt hơn khi so sánh với các phương pháp đơn.

Những đánh giá trên có thể được làm rõ tại Hình 3.7, ở đây minh hoạ đường cong ROC của mỗi phương pháp đơn và phương pháp OFuseAD trên một số bộ dữ liệu cụ thể (UNSW-NB15, CTU13_08, CTU13_13). Nó cho ta thấy rằng đỉnh đường cong ROC của mô hình OFuseAD luôn có đỉnh hướng tới giá trị (0,1), ngay cả khi giá trị AUC (trên tập CTU13_13) của nó không lớn nhất khi so sánh với các phương pháp đơn. Điều đấy thể hiện tính hiệu quả trong khả năng phát hiện bất thường của phương pháp tổng hợp so với phương pháp đơn.

Khi xét đơn vị đo AUC như trên Bảng 3.2, mô hình OFuseAD có vẻ như cho kết quả AUC không tốt so với các phương pháp đơn. Tuy nhiên điều này không hoàn toàn đúng vậy, bởi vì trong điều kiện ứng dụng thực tiễn (real-world application), các mô hình phát hiện bất thường được yêu cầu phải có một ngưỡng quyết định [18], [74], vì thế giá trị AUC sẽ thể hiện không rõ hiệu quả của việc so sánh giữa phương pháp tổng hợp và các phương pháp đơn phát hiện bất thường khi yêu cầu phải cung cấp thông tin ở mức cụ thể hơn, là nhãn nhị phân [69]. Trong ngữ cảnh này, các đơn vị đo F1-Score là chủ yếu và kết quả theo chỉ số này cho mô hình OFuseAD đã được chứng minh tốt so với các phương pháp đơn như đã thảo luận ở trên.

Hơn thế nữa, mô hình xây dựng trên OFuseAD cho phép hệ thống tự xác định ngưỡng ra quyết định, không giống như với các mô hình NAD xây dựng bởi phương pháp OCC khác. Với bài toán NAD trong thực tiễn, việc cung cấp kết quả là bất thường hay bình thường đối với một tình huống an ninh mạng là cần thiết. Trong bài toán NAD dựa trên OCC, xác định ngưỡng được cho là nhiệm vụ rất khó ngay cả với các chuyên gia [20], [40]. Kết quả thực nghiệm thể



Hình 3.7: *Minh hoạ đường cong ROC và giá trị AUC*

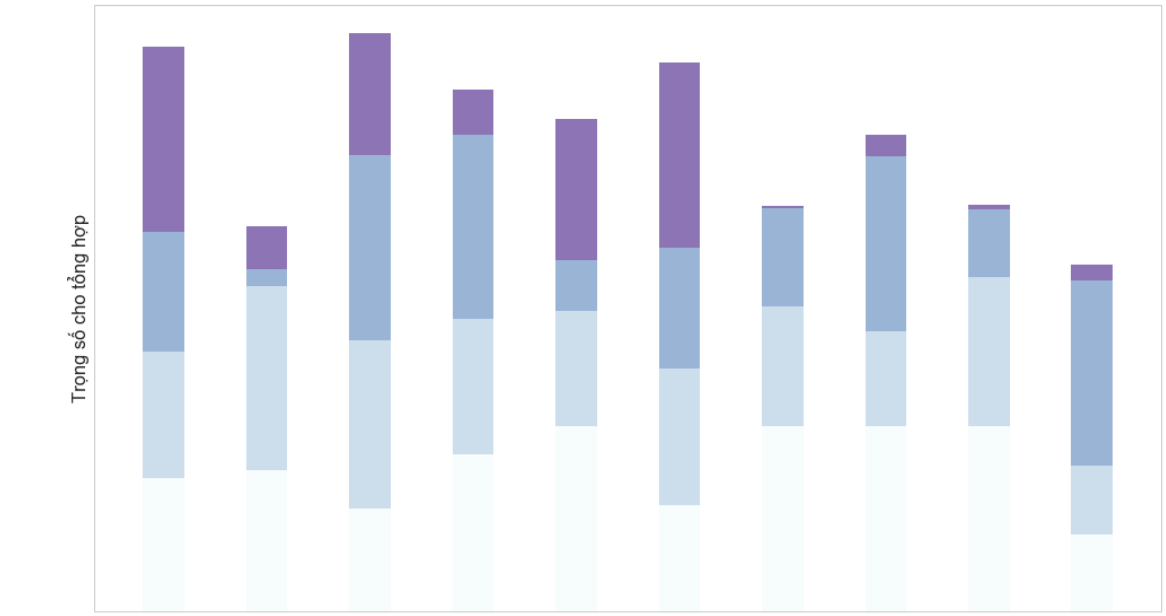
hiện, mô hình phát hiện bất thường mạng được xây dựng từ OFuseAD có khả năng tự thiết lập ngưỡng quyết định, giúp mô hình có khả năng cung cấp thông tin cụ thể hơn, đó là nhãn nhị phân (1 ứng với bất thường và 0 ứng với bình thường), như thảo luận tại phần 1.1.4.

Kết quả thực nghiệm theo chỉ số F1-score, ACC đã thể hiện tính hiệu quả của mô hình đề xuất, kết quả đó cũng thể hiện khả năng của giải pháp trong việc tự thiết lập trọng số cho các SglAD tham gia vào mô hình DF. Hình 3.8 thể hiện kết quả trọng số tham gia tổng hợp của các OCC trên các tập dữ liệu khác nhau. Chi tiết hơn, tại Bảng 3.5, thể hiện độ đo "sinh lỗi" và trọng số tham gia tổng hợp của các OCC trên một bộ dữ liệu cụ thể.

Như đã đề cập, kết quả thực nghiệm được thực hiện với các giá trị *bw* khác nhau với mục đích để phân tích sự ảnh hưởng của tham số này đối với kết quả

Bảng 3.5: *Độ đo "sinh lỗi" và trọng số các OCC tham gia mô hình tổng hợp (CTU13_09)*

Phương pháp	**Lỗi ở mỗi ngưỡng**									**Trung bình**	**Trọng số**
	90%	**91%**	**92%**	**93%**	**94%**	**95%**	**96%**	**97%**	**98%**		
DSAE_Z1	0.000	0.001	0.002	0.001	0.000	0.00	0.001	0.003	0.003	0.00128	1.000
DSAE_Z2	0.001	0.003	0.000	0.000	0.001	0.001	0.004	0.003	0.004	0.00206	0.622
LOF	0.008	0.006	0.004	0.003	0.006	0.005	0.005	0.006	0.003	0.00461	0.277
KDE	0.001	0.001	0.000	0.001	0.000	0.001	0.002	0.003	0.005	0.00167	0.767



	SCADA	IoT	CTU13_10	CTU13_08	CTU13_09	CTU13_13	Spambase	UNSW	NSLKDD	InternetAds
KDE	1.00000	0.23106	0.66129	0.24444	0.76667	1.00000	0.01072	0.11377	0.02430	0.08265
LOF	0.65079	0.09242	1.00000	1.00000	0.27711	0.65385	0.53333	0.95000	0.36420	1.00000
DSAE_Z1	0.68333	1.00000	0.91111	0.73333	0.62162	0.73913	0.64516	0.51351	0.80822	0.37249
DSAE_Z2	0.71930	0.75934	0.55405	0.84615	1.00000	0.57303	1.00000	1.00000	1.00000	0.41629

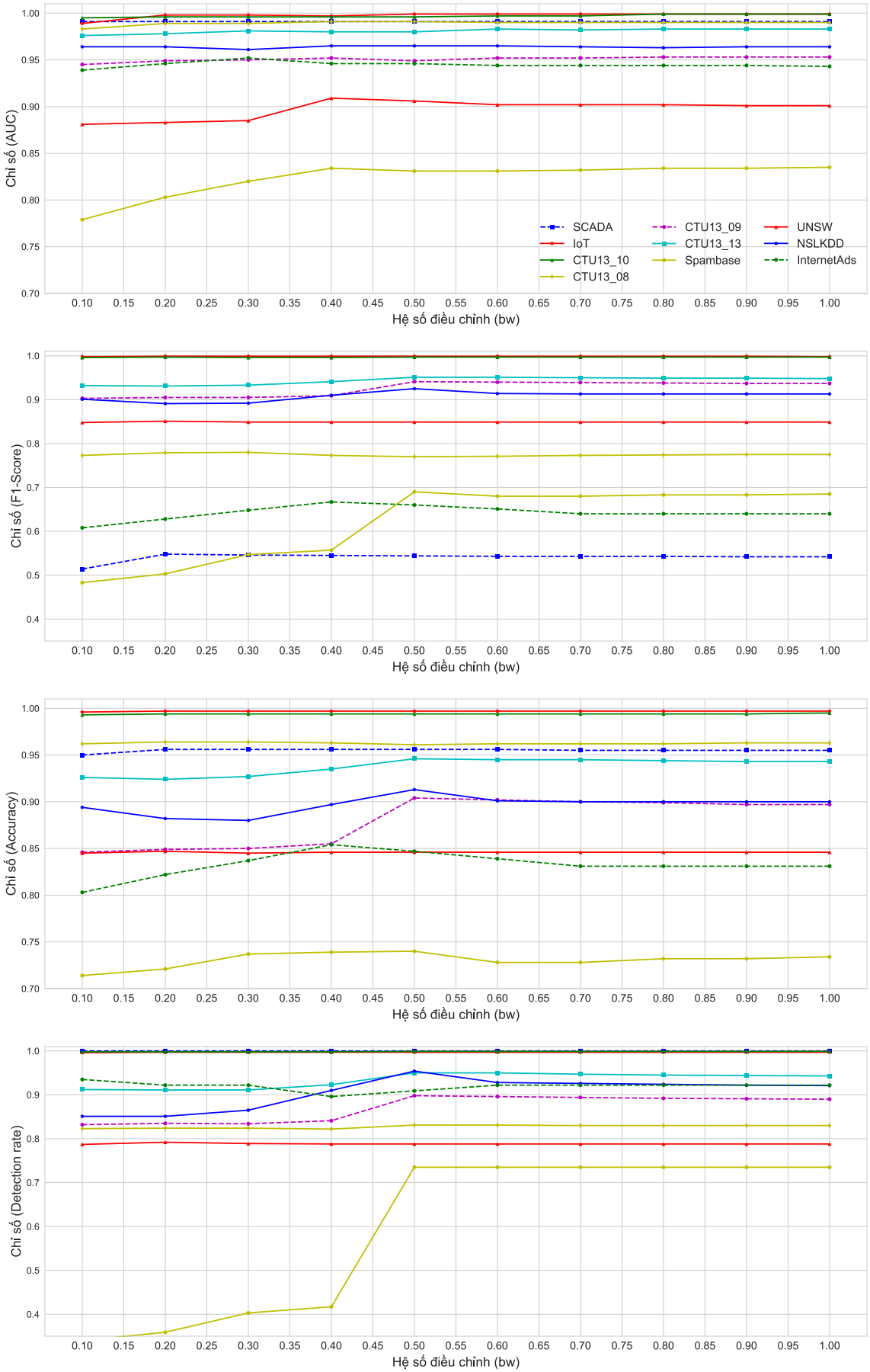
Hình 3.8: *Trọng số tham gia tổng hợp của các OCC được tính cho mười tập dữ liệu*

của mô hình đề xuất. Hình 3.9 cũng cho thấy rằng, việc thay đổi giá trị của tham số này không ảnh hưởng nhiều đến kết quả tổng thể, khi minh hoạ các đơn vị đo (gồm: AUC, F1-Score, ACC, DR) trên biểu đồ theo sự thay đổi giá trị của bw trong khoảng $(0, 1)$, các đơn vị này cho giá trị gần như ổn định khi bw vượt qua giá trị 0.5. Điều này cho kết quả phù hợp theo cách đã định nghĩa bw như đã đề cập trước đây, khi bw ở giá trị tối thiểu 0.5, giá trị đầu tiên trong vùng giá trị khuyến nghị, mô hình hoạt động cho thấy hiệu quả trên các tập dữ liệu đã kiểm thử.

Quá trình thực nghiệm, các kịch bản khác nhau cho các bộ phân đơn lớp trong mô hình OFuseAD như: sử dụng riêng lẻ SAE, DSAE_Z1, DSAE_Z2 lần lượt đại diện cho bộ phân đơn lớp dựa trên học sâu. Kết quả cũng cho thấy mô hình NAD được xây dựng hoạt động khả thi, ổn định và hiệu quả tương đối tốt so với các phương pháp đơn trên đa số các tập dữ liệu được thực nghiệm. Tuy nhiên, các kết quả này đều không tốt hơn phương án sử dụng cả DSAE_Z1, DSAE_Z2 cho OFuseAD. Điều này cũng phù hợp với nhận định đã được đưa ra ở Chương 2, đó là việc kết hợp cả DSAE_Z1 và DSAE_Z2 có tiềm năng cho xây dựng một mô hình phát hiện bất thường hiệu quả hơn.

Kết quả thực nghiệm như tại Bảng 3.3 cho thấy phương pháp OFuseAD có lợi thế rõ hơn đối với các tập dữ liệu mà việc phân tách giữa bình thường và bất thường khó. Cụ thể, với các bộ dữ liệu SCADA, CTU13_08, Spambase, InternetADs thì các phương pháp đơn cơ bản đều cho kết quả rất thấp, kết quả theo chỉ số F1-score của phương pháp tổng hợp cải tiến chỉ số F1-score đạt trung bình là 15% so với trung bình các phương pháp đơn (có giá trị F1-score trung bình tương ứng 0.421, 0.737, 0.401, 0.508 của các phương pháp đơn trên tương ứng 04 tập dữ liệu SCADA, CTU13_08, Spambase, InternetADs). Ngược lại, với các bộ dữ liệu được phân tách tốt (như IoT, CTU13_10, CTU13_13, thể hiện ở đa số kết quả phương pháp đơn đều có F1-score khá tương đồng và đạt trên 0.9x), thì giá trị cải tiến này chỉ đạt ở 2.4%.

Mục tiêu của luận án tập trung vào cải tiến khả năng phát hiện bất thường



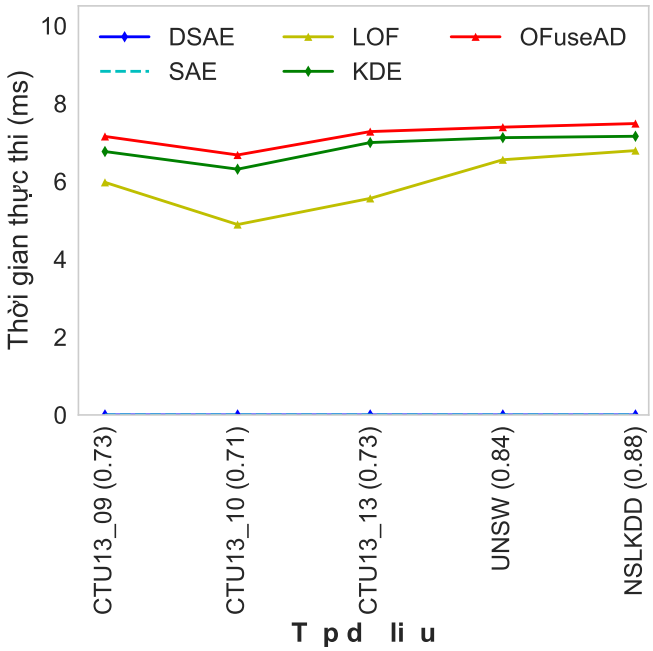
Hình 3.9: *Ảnh hưởng bw đến hiệu quả của OFuseAD.*

trên phương diện độ chính xác và tính ổn định. Do vậy, phạm vị luận án không đi sâu đến vấn đề độ phức tạp thuật toán, cụ thể là độ phức tạp trong huấn luyện các phương pháp học máy, học sâu. Còn độ phức tạp trong quá trình kiểm tra của phương pháp tổng hợp dữ liệu đề xuất và các phương pháp đơn phát hiện bất thường có thể được đánh giá như sau.

Trước tiên, xét độ phức tạp của từng thuật toán đơn phát hiện bất thường khi xử lý n mẫu dữ liệu kiểm tra, với DSAE (là mô hình học sâu dựa trên AutoEncoder) có giá trị tương ứng là $\mathcal{O}(1)$ [21], LOF tương ứng $\mathcal{O}(n^2)$ [7] và KDE tương ứng $\mathcal{O}(n^2)$ [94]. Độ phức tạp thuật toán của khối xử lý D-S như đã đề cập ở mục 1.3.4.2 lên đến $\mathcal{O}(n * 2^{|U|})$, tuy nhiên trong trường hợp ứng dụng, mà cụ thể là với OFuseAD, $\Theta = (A, N)$, không tính tập rỗng, giá trị độ phức tạp có thể xem là $\mathcal{O}(n * 2^{|U|})$=$\mathcal{O}(n * 2^3)$=$\mathcal{O}(n)$. Như vậy, phương pháp phát hiện bất thường dựa trên tổng hợp dữ liệu đề xuất có độ phức tạp $\mathcal{O}(1)$+ $\mathcal{O}(n^2)$+$\mathcal{O}(n^2)$+$\mathcal{O}(n)$, và tương đương với $\mathcal{O}(n^2)$.

Quá trình thực nghiệm để so sánh thời gian xử lý đối với một mẫu dữ liệu (query time), luận án thực hiện đo thời gian thực thi của SAE, DSAE, LOF, KDE và mô hình OFuseAD theo phương pháp sau. Tạo các tập dữ liệu con bằng cách lấy ngẫu nhiên 1000 mẫu dữ liệu tương ứng của năm tập dữ liệu kiểm tra của CTU13_09,CT13_10, CTU13_13, UNSW-NB15, NSLKDD. Mục đích của việc lấy mẫu để tạo ra các bộ dữ liệu kiểm tra có số lượng mẫu như nhau, giúp cho việc đối chiếu kết quả được tường minh hơn. Thực nghiệm kiểm tra 100 lần đối với từng tập dữ liệu con này bằng mô hình SAE, DSAE, LOF, KDE và mô hình OFuseAD với tham số thiết lập như đã mô tả ở phần trên. Tính toán thời gian truy vấn cho mỗi mẫu dữ liệu. Kết quả thời gian kiểm tra của mỗi mô hình trên năm tập dữ liệu được mô tả tại Hình 3.10.

Đường màu đỏ thể hiện thời gian xử lý của phương pháp được đề xuất khi thực nghiệm trên năm tập dữ liệu khác nhau, kết quả cho thấy thời gian xử lý của mô hình NAD được xây dựng từ OFFuseAD chỉ nhiều hơn thời gian xủ lý của các phương pháp đơn KDE và LOF không đáng kể, chỉ lớn hơn thời gian



Hình 3.10: *Thời gian truy vấn của các phương pháp khác nhau*

xử lý của KDE khoảng 0.5%.

3.5. Kết luận

Nội dung nghiên cứu trong chương đã giải quyết vấn đề thứ hai và ba trong phát biểu bài toán luận án, bao gồm. Thứ nhất, khắc phục được hạn chế của một phương pháp đơn được cho là thường chỉ tốt trên bài toán (tập dữ liệu) cụ thể mà không thực sự hiệu quả trên các bài toán khác. Thứ hai, yêu cầu giải pháp NAD có khả năng tự thiết lập ngưỡng quyết định, các phương pháp NAD theo hướng OCC được cho là vẫn phụ thuộc nhiều vào chuyên gia trong việc thiết lập ngưỡng để có thể cung cấp kết quả đầu ra nhãn nhị phân, nghĩa là một mẫu dữ liệu đang quan sát là bình thường hay bất thường.

Kết quả nghiên cứu đã đề xuất được phương pháp khung NAD dựa trên tổng hợp dữ liệu, có tên OFuseAD. OFuseAD áp dụng hiệu quả lý thuyết D-S để kết hợp lợi thế từ các phương pháp đơn OCC truyền thống và học sâu. Đây

là nội dung nghiên cứu với nhiều vấn đề cần giải quyết như: đưa ra cơ sở cho lựa chọn các phương pháp đơn OCC tham gia mô hình; xác định ngưỡng quyết định cho từng phương pháp đơn; phương pháp xác định độ tin cậy (trọng số) giữa các phương pháp đơn tham gia mô hinh DF; ngoài ra, các vấn đề của lý thuyết D-S đã được nghiên cứu đề xuất ứng dụng vào mô hình như, định nghĩa tập giả thuyết cho hệ thống (FoD), xây dựng hàm gán trọng số niềm tin (BPA) theo đặc thù bài toán phát hiện bất thường, và mấu chốt là đã đề xuất được giải pháp cải tiến DRC cho phù hợp với bài toán tổng hợp các nguồn là các bộ phân đơn lớp (OCC).

Quá trình thực nghiệm, mô hình NAD được xây dựng từ mô hình khung OFuseAD sử dụng các phương pháp đơn OCC gồm: mạng nơ-ron học sâu, sử dụng DSAE; phát hiện dựa trên khoảng cách, sử dụng LOF; phát hiện dựa trên mật độ, sử dụng KDE. Kết quả thực nghiệm được thực hiện trên mười tập dữ liệu phổ biến, tiên tiến trong lĩnh vực an ninh mạng cho thấy. OFuseAD hoạt động khả thi và có hiệu quả, mô hình NAD dựa trên phương pháp khung OFuseAD cho độ ổn định và khả năng phát hiện bất thường hiệu quả hơn các phương pháp đơn OCC trên hầu hết (9 trên 10) các bộ dữ liệu được thực nghiệm. Kết quả cũng cho thấy, giải pháp đề xuất cho phép hoạt động với ngưỡng quyết định được tự động thiết lập, giúp cho mô hình NAD đề xuất đáp ứng yêu cầu thực tiễn. Phương pháp OFuseAD có xu hướng lợi thế hơn đối với các bộ dữ liệu khó. Đây là các bộ dữ liệu hiện hữu với độ phân tách bình thường và bất thường không tốt.

Nội dung trong chương cũng chỉ ra, độ phức tạp trong truy vấn của mô hình khung đề xuất cơ bản phụ thuộc vào việc lựa chọn các phương pháp đơn (bằng $\mathcal{O}(n)+\mathcal{O}(n^2)$, trong đó $\mathcal{O}(n^2)$ là độ phức tạp của các phương pháp đơn như KDE và LOF). OFuseAD là mô hình khung, do vậy việc lựa chọn phương pháp đơn là hoàn toàn độc lập theo từng bài toán cụ thể. Tuy nhiên, việc OFuseAD sử dụng các phương pháp đơn dựa trên khoảng cách và dựa trên mật độ thường cho độ phức tạp lớn, do vậy OFuseAD phải chịu trả giá cho độ phức tạp tính toán.

Bài toán xây dựng mô hình khung NAD dựa trên tổng hợp dữ liệu, sử dụng lý thuyết D-S để kết hợp được lợi thế từ các bộ phân đơn lớp OCC cả học sâu và truyền thống có tính mới. Trong hiểu biết của nghiên cứu sinh, chưa có nghiên cứu tương tự được thực hiện.

KẾT LUẬN

Như vậy, luận án đã nghiên cứu và giải quyết các vấn đề theo phát biểu bài toán đặt ra ban đầu khi thực hiện cải tiến phương pháp phát hiện bất thường mạng. Các kết quả nghiên cứu đã được công bố trong các công trình khoa học uy tín trong và ngoài nước. Nội dung của luận án được trình bày dựa theo phương pháp nghiên cứu đã đặt ra.

Trong phần mở đầu, luận án tập trung trình bày làm rõ vấn đề khoa học cần giải quyết, gồm ba vấn đề chính: (1) cải tiến một số hạn chế cơ bản của phương pháp tiêu biểu cho phát hiện bất thường mạng dựa trên học sâu; (2) đề xuất giải pháp khắc phục hạn chế chung đối với phương pháp đơn cho phát hiện bất thường (SglAD). Mỗi phương pháp đơn thường chỉ tốt trên bài toán (tập dữ liệu) cụ thể mà thường không tốt trên các bài toán khác; (3) phương pháp phát hiện bất thường cần tự động thiết lập ngưỡng ra quyết định. Việc thiết lập ngưỡng sẽ giúp cho mô hình xác định, cung cấp thông tin cụ thể hơn, qua đó có thể triển khai các giải pháp phát hiện bất thường vào ứng dụng thực tế.

Trong chương thứ nhất, luận án trình bày các nội dung cơ sở liên quan đến luận án, tập trung vào làm rõ về khái niệm, mô hình tổng thể phát hiện bất thường mạng, các thành phần chính của mô hình. Chương này cũng giới thiệu một số kết quả nghiên cứu liên quan, gồm: một số các nghiên cứu về phương pháp đơn cho phát hiện bất thường; một số các nghiên cứu về tổng hợp, kết hợp dữ liệu ra quyết định. Nội dung chương cũng giới thiệu một số bộ dữ liệu và chỉ số cho kiểm thử, đánh giá các phương pháp phát hiện bất thường mạng. Kết quả nghiên cứu liên quan được công bố trên các công trình khoa học [CT4]. Trong chương thứ hai, luận án trình bày kết quả nghiên cứu để giải quyết vấn đề thứ nhất mà luận án đã đặt ra trong phát biểu bài toán. Đã đề xuất được giải pháp cho khắc phục hai thách thức mà phương pháp NAD tiêu biểu dựa trên học sâu

đang gặp phải. Kết quả nghiên cứu liên quan được công bố trên các công trình khoa học [CT1], [CT5]. Chương thứ ba, luận án trình bày kết quả nghiên cứu có tính mới và phức tạp hơn, nội dung nghiên cứu trong chương giải quyết hai vấn đề còn lại của luận án. Theo đó, kết quả trong chương đã chứng minh lý thuyết D-S rất phù hợp cho bài toán phát hiện bất thường, luận án đã đề xuất được phương pháp có tính khung theo hướng kết hợp nhiều phương pháp đơn OCC, để tạo ra phương pháp tổng thể có khả năng phát hiện bất thường mạnh hơn, có độ chính xác và tính ổn định cao hơn, mô hình khung đề xuất có tên là OFuseAD. Ngoài ra, mô hình NAD dựa trên tổng hợp dữ liệu đã đề xuất còn có khả năng tự động ước lượng ngưỡng ra quyết định. Kết quả nghiên cứu liên quan được công bố trên các công trình khoa học [CT2], [CT3], [CT6].

Một số đóng góp chính của luận án, các hạn chế cũng như định hướng nghiên cứu tương lai được trình bày trong phần tiếp theo.

1. Một số kết quả chính của luận án

- Luận án đề xuất được các mô hình phát hiện bất thường sử dụng theo mạng nơ-ron học sâu có tên Clustering-Shrink AutoEncoder và Double-Shrink AutoEncoder (DSAE). Trong đó, DSAE là mô hình NAD mới và có hướng đi khác với các giải pháp mạng nơ-ron học sâu cho lĩnh vực phát hiện bất thường đã công bố khi sử dụng đồng thời cả hai yếu tố là RE và vector lớp ẩn làm cơ sở đưa ra độ đo bất thường. Kết quả thực nghiệm đã cho thấy, DSAE có thể phát hiện hiệu quả hơn với các tấn công mà mô hình tiêu biểu SAE gặp khó. Các tấn công này được cho là có dữ liệu rất giống với dữ liệu bình thường, do vậy thường tạo ra khó khăn với các phương pháp đã có.

- Luận án đã đề xuất được một phương pháp có tính khung cho giải quyết các hạn chế được cho là hiện hữu với các phương pháp phát hiện bất thường đơn lẻ, mô hình có tên là OFuseAD. Theo đó, OFuseAD cho phép xây dựng các

mô hình phát hiện bất thường từ các phương pháp phân đơn lớp (One-class Classification - OCC). Thêm vào đó, giải pháp này không cần sự can thiệp của chuyên gia trong thiết lập ngưỡng quyết định mà vẫn cung cấp được thông tin cụ thể ở mức nhãn nhị phân.

- Luận án đã đề xuất giải pháp cụ thể trong ứng dụng lý thuyết Dempster-Shafer (D-S) cho bài toán OCC. Đây là lý thuyết mạnh và đang được quan tâm bởi nhiều nhà nghiên cứu trên thế giới; tuy nhiên ở Việt Nam, hiện chưa thấy nhiều công bố các nghiên cứu sâu về lý thuyết này. Hai đóng góp cụ thể lớn nhất khi áp dụng lý thuyết này trong luận án là: đề xuất xây dựng hàm BPA theo đặc thù bài toán phát hiện bất thường; đề xuất được hàm DRC_AD, đây là giải pháp mở rộng của hàm kết hợp DRC của lý thuyết D-S, việc mở rộng này giúp cho lý thuyết D-S thực tiễn hơn. Vì DRC nguyên bản xem các nguồn có độ tin cậy như nhau nhưng thực tế các nguồn thường có độ tin cậy khác nhau.

2. Một số giới hạn của luận án

Bên cạnh các kết quả đã đạt được, luận án vẫn còn một số hạn chế, một trong số đó là việc giả định các nguồn cung cấp thông tin (các phương pháp đơn) trong OFuseAD đều đang quan sát cùng một đối tượng thông tin gốc như nhau. Trong thực tế vẫn có nhiều bài toán, việc nhiều nguồn thông tin gốc khác nhau nhưng đều tham gia đóng góp cho cùng một giả định của hệ thống.

Thêm vào đó, độ phức tạp tính toán của OFuseAD phụ thuộc lớn vào các phương pháp đơn, trong khi OFuseAD sử dụng các phương pháp đơn OCC dựa trên khoảng cách và dựa trên mật độ, các phương pháp này thường cho chi phí tính toán rất lớn.

3. Hướng nghiên cứu trong tương lai

Một số hướng nghiên cứu mở rộng, phát triển kết quả của luận án có thể thực hiện trong tương lai như: Đầu tiên, tiếp tục nghiên cứu cải tiến mô hình DSAE để có thể áp dụng cho các bài toán phát hiện bất thường khác, không chỉ dừng lại ở lĩnh vực an ninh mạng như trong luận án. Thêm vào đó, việc thử nghiệm trên phạm vi rộng hơn, sử dụng trên môi trường mạng thật, hoặc áp dụng DSAE cho một vùng mạng có tính rất đặt thù để đánh giá kỹ hơn hiệu quả thuật toán đề xuất.

Thứ hai, nghiên cứu mở rộng OFuseAD theo hướng sử dụng các nguồn thông tin gốc khác nhau cho các phương pháp đơn. Nghiên cứu xây dựng các mô hình phát hiện bất thường dựa trên OFuseAD cho các lĩnh vực khác.

Thứ ba, trên cơ sở kết quả mở rộng hàm kết hợp DRC của lý thuyết D-S. Phát triển việc ứng dụng lý thuyết này cho các lĩnh vực khác, đặc biệt là các bài toán liên quan đến phân lớp, phân cụm cũng như xác định các đối tượng có tính mới, lạ./.

CÁC CÔNG TRÌNH LIÊN QUAN ĐẾN LUẬN ÁN

I. HỘI THẢO QUỐC TẾ:

[CT1] **Thanh Cong Bui**, Loi Van Cao, Minh Hoang, and Quang Uy Nguyen. A clustering-based shrink autoencoder for detecting anomalies in intrusion detection systems. *In 2019 11th International Conference on Knowledge and Systems Engineering (KSE)*, pp. 1–5. IEEE, (2019).

[CT2] **Thanh Cong Bui**, Minh Hoang, Quang Uy Nguyen, and Cao Loi Van. Data fusion-based network anomaly detection towards evidence theory. *2019 6th NAFOSTED International Conference on Information and ComputerScience (NICS'19)*. pp. 33–38. IEEE (2019) (Được trao giải bài báo tốt nhất (The Best Paper Award)).

II. TẠP CHÍ TRONG NƯỚC:

[CT3] **Bùi Công Thành**, Vũ Tuấn Anh, Hoàng Trung Kiên. Ứng dụng lý thuyết Dempster Shafer trong xây dựng mô hình suy luận. *Tạp chí Nghiên cứu Khoa học và Công nghệ Quân sự*, 50(08) 08.2017, 144–157 (2017).

[CT4] **Bùi Công Thành**, Nguyễn Quang Uy, Hoàng Minh. Một số bộ dữ liệu kiểm thử phổ biến cho phát hiện xâm nhập mạng và đặc tính phân cụm. *Tạp chí Khoa học và Công nghệ Việt Nam, Bộ Khoa học và Công nghệ*, 62(1) 1.2020:1–7, (2020), (Series B), ISSN 1859-4794.

[CT5] **Thanh Cong Bui**, Loi Van Cao, Minh Hoang, and Quang Uy Nguyen. *Double-shrink autoencoder for network anomaly detection. Tạp chí Tin học điều khiển, Viện Hàn lâm Khoa học và Công nghệ Việt Nam* V.36, N.2 (2020).

III. TẠP CHÍ QUỐC TẾ:

[CT6] **Thanh Cong Bui**, Van Loi Cao, Quang Uy Nguyen, and Minh Hoang. One-class Fusion-based Learning Model for Anomaly Detection. *Journal of Com-*

puter in Industry: Classification, Machine learning, pp. ...–.... (ISI-SCIE, IF=3.954)(2021) (Under Review).

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 Ngọc Thanh, Trần Văn Lăng và Hoàng Tùng (2016), "Một tiếp cận máy học để phân lớp các kiểu tấn công trong hệ thống phát hiện xâm nhập mạng", *Kỷ yếu Hội nghị Khoa học Quốc gia lần thứ IX - Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR'9)*, 10.15625/vap.2016.00061, tr. 502–508.

Tiếng Anh:

[3] Iftikhar Ahmad, Azween B Abdullah, and Abdullah S Alghamdi, "Remote to Local attack detection using supervised neural network", in: *2010 International Conference for Internet Technology and Secured Transactions*, IEEE, 2010, pp. 1–6.

[4] 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.

[5] 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.

[6] Bahnsen Alejandro (2016), "Correa", *Building ai applications using deep learning*.

[7] Malak Alshawabkeh, Byunghyun Jang, and David Kaeli, "Accelerating the local outlier factor algorithm on a GPU for intrusion detection systems", in: *Proceedings of the 3rd Workshop on General-Purpose Computation on Graphics Processing Units*, 2010, pp. 104–110.

[8] Fabrizio Angiulli and Clara Pizzuti, "Fast outlier detection in high dimensional spaces", in: *European Conference on Principles of Data Mining and Knowledge Discovery*, Springer, 2002, pp. 15–27.

[9] Arthur Asuncion and David Newman, *UCI machine learning repository*, 2007.

[10] Tim Bass (2000), "Intrusion detection systems and multisensor data fusion: Creating cyberspace situational awareness", *Communications of the ACM*, 43 (4), pp. 99–105.

[11] Pavel Berkhin, "A survey of clustering data mining techniques", in: *Grouping multidimensional data*, Springer, 2006, pp. 25–71.

[12] Dhruba Kumar Bhattacharyya and Jugal Kumar Kalita (2013), *Network anomaly detection: A machine learning perspective*, Crc Press.

[13] Monowar H Bhuyan, Dhruba Kumar Bhattacharyya, and Jugal K Kalita (2013), "Network anomaly detection: methods, systems and tools", *Ieee communications surveys & tutorials*, 16 (1), pp. 303–336.

[14] Monica Bianchini and Franco Scarselli, "On the complexity of shallow and deep neural network classifiers.", in: *ESANN*, Citeseer, 2014.

[15] Hervé Bourlard and Yves Kamp (1988), "Auto-association by multilayer perceptrons and singular value decomposition", *Biological cybernetics*, 59 (4-5), pp. 291–294.

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

[17] Van Loi Cao (2018), "Improving Network Anomaly Detection with Genetic Programming and Autoencoders".

[18] Van Loi Cao, Miguel Nicolau, and James McDermott, "A hybrid autoencoder and density estimation model for anomaly detection", in: *International Conference on Parallel Problem Solving from Nature*, Springer, 2016, pp. 717–726.

[19] Van Loi Cao, Miguel Nicolau, and James McDermott, "One-class classification for anomaly detection with kernel density estimation and genetic programming", in: *European Conference on Genetic Programming*, Springer, 2016, pp. 3–18.

[20] Van Loi Cao, Miguel Nicolau, and James McDermott (2019), "Learning Neural Representations for Network Anomaly Detection.", *IEEE transactions on cybernetics*, 49 (8), pp. 3074–3087.

[21] Raghavendra Chalapathy and Sanjay Chawla (2019), "Deep Learning for Anomaly Detection: A Survey", *arXiv*, arXiv–1901.

[22] Varun Chandola, Arindam Banerjee, and Vipin Kumar (2009), "Anomaly Detection: A Survey", *ACM Comput. Surv.*, 41 (3), 15:1–15:58, ISSN: 0360-0300, DOI: 10.1145/1541880.1541882, URL: http://doi.acm.org/10.1145/1541880.1541882.

[23] Vassilis Chatzigiannakis and Symeon Papavassiliou (2007), "Diagnosing anomalies and identifying faulty nodes in sensor networks", *IEEE Sensors Journal*, 7 (5), pp. 637–645.

[24] Qi Chen and Uwe Aickelin (2006), "Anomaly detection using the Dempster-Shafer method", *Available at SSRN 2831339*.

[25] Qi Chen et al. (2014), "Data classification using the Dempster–Shafer method", *Journal of Experimental & Theoretical Artificial Intelligence*, 26 (4), pp. 493–517.

[26] Thomas M Chen and Varadharajan Venkataramanan (2005), "Dempster-Shafer theory for intrusion detection in ad hoc networks", *IEEE Internet Computing*, 9 (6), pp. 35–41.

[27] Gillian Cleary, *ISTR (Internet Security Threat Report).*

[28] Elisa Costante et al., "A hybrid framework for data loss prevention and detection", in: *2016 IEEE Security and Privacy Workshops (SPW)*, IEEE, 2016, pp. 324–333.

[29] Dipankar Dasgupta and Nivedita Sumi Majumdar, "Anomaly detection in multidimensional data using negative selection algorithm", in: *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600)*, vol. 2, IEEE, 2002, pp. 1039–1044.

[30] Dipankar Dasgupta and Fernando Nino, "A comparison of negative and positive selection algorithms in novel pattern detection", in: *Smc 2000 conference proceedings. 2000 ieee international conference on systems, man and cybernetics.'cybernetics evolving to systems, humans, organizations, and their complex interactions'(cat. no. 0*, vol. 1, IEEE, 2000, pp. 125–130.

[31] Remco C De Boer (2002), "A Generic architecture for fusion-based intrusion detection systems".

[32] L Dhanabal and SP Shantharajah (2015), "A study on NSL-KDD dataset for intrusion detection system based on classification algorithms", *International Journal of Advanced Research in Computer and Communication Engineering*, 4 (6), pp. 446–452.

[33] Luca Didaci, Giorgio Giacinto, and Fabio Roli, "Ensemble learning for intrusion detection in computer networks", in: *Workshop Machine Learning Methods Applications, Siena, Italy*, 2002.

[34] A Dissanayake (2008), "Intrusion Detection Using the Dempster-Shafer Theory. 60-510 Literature Review and Survey", *School of Computer Science, University of Windsor*.

[35] Abhishek Divekar et al., "Benchmarking datasets for anomaly-based network intrusion detection: KDD CUP 99 alternatives", in: *2018 IEEE 3rd International Conference on Computing, Communication and Security (ICCCS)*, IEEE, 2018, pp. 1–8.

[36] Ke-Lin Du and MNS Swamy, "Combining Multiple Learners: Data Fusion and Ensemble Learning", in: *Neural Networks and Statistical Learning*, Springer, 2019, pp. 737–767.

[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] Nabila Farnaaz and MA Jabbar (2016), "Random forest modeling for network intrusion detection system", *Procedia Computer Science*, 89 (1), pp. 213–217.

[39] Gilberto Fernandes et al. (2019), "A comprehensive survey on network anomaly detection", *Telecommunication Systems*, 70 (3), pp. 447–489.

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

[41] Ugo Fiore et al. (2013), "Network anomaly detection with the restricted Boltzmann machine", *Neurocomputing*, 122, pp. 13–23.

[42] Sebastian Garcia et al. (2014), "An empirical comparison of botnet detection methods", *computers & security*, 45, pp. 100–123.

[43] Pedro Garcia-Teodoro et al. (2009), "Anomaly-based network intrusion detection: Techniques, systems and challenges", *computers & security*, 28 (1-2), pp. 18–28.

[44] Amol Ghoting, Srinivasan Parthasarathy, and Matthew Eric Otey (2008), "Fast mining of distance-based outliers in high-dimensional datasets", *Data Mining and Knowledge Discovery*, 16 (3), pp. 349–364.

[45] Giorgio Giacinto, Fabio Roli, and Luca Didaci (2003), "Fusion of multiple classifiers for intrusion detection in computer networks", *Pattern recognition letters*, 24 (12), pp. 1795–1803.

[46] Xavier Glorot and Yoshua Bengio, "Understanding the difficulty of training deep feedforward neural networks", in: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.

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

[48] Ian Goodfellow, Yoshua Bengio, and Aaron Courville (2016), *Deep learning*, MIT press.

[49] David L Hall and James Llinas (1997), "An introduction to multisensor data fusion", *Proceedings of the IEEE*, 85 (1), pp. 6–23.

[50] Ville Hautamaki, Ismo Karkkainen, and Pasi Franti, "Outlier detection using k-nearest neighbour graph", in: *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.* Vol. 3, IEEE, 2004, pp. 430–433.

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

[52] Simon Hawkins et al., "Outlier detection using replicator neural networks", in: *International Conference on Data Warehousing and Knowledge Discovery*, Springer, 2002, pp. 170–180.

[53] 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.

[54] Wei Hu, Jianhua Li, and Qiang Gao, "Intrusion detection engine based on Dempster-Shafer's theory of evidence", in: *2006 International Conference on Communications, Circuits and Systems*, vol. 3, IEEE, 2006, pp. 1627–1631.

[55] Nathalie Japkowicz, Catherine Myers, Mark Gluck, et al., "A novelty detection approach to classification", in: *IJCAI*, vol. 1, 1995, pp. 518–523.

[56] P Gifty Jeya, M Ravichandran, and CS Ravichandran (2012), "Efficient classifier for R2L and U2R attacks", *International Journal of Computer Applications*, 45 (21), pp. 28–32.

[57] Jayakumar Kaliappan, Revathi Thiagarajan, and Karpagam Sundararajan (2015), "Fusion of heterogeneous intrusion detection systems for network attack detection", *The Scientific World Journal*, 2015.

[58] Alexandros Kaltsounidis and Isambo Karali, "Dempster-Shafer Theory: How Constraint Programming Can Help", in: *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, Springer, 2020, pp. 354–367.

[59] Bahador Khaleghi et al. (2013), "Multisensor data fusion: A review of the state-of-the-art", *Information fusion*, 14 (1), pp. 28–44.

[60] Yoon Kim (2014), "Convolutional neural networks for sentence classification", *arXiv preprint arXiv:1408.5882*.

[61] Nickolaos Koroniotis et al. (2019), "Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-iot dataset", *Future Generation Computer Systems*, 100, pp. 779–796.

[62] Roshan Kumar and Deepak Sharma, "HyINT: signature-anomaly intrusion detection system", in: *2018 9th International Conference on*

Computing, Communication and Networking Technologies (ICCCNT), IEEE, 2018, pp. 1–7.

[63] Donghwoon Kwon et al. (2017), "A survey of deep learning-based network anomaly detection", *Cluster Computing*, pp. 1–13.

[64] Twan van Laarhoven (2017), "L2 Regularization versus Batch and Weight Normalization", *arXiv*, arXiv–1706.

[65] Pavel Laskov et al. (2004), "Intrusion detection in unlabeled data with quarter-sphere support vector machines", *Praxis der Informationsverarbeitung und Kommunikation*, 27 (4), pp. 228–236.

[66] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton (2015), "Deep learning", *nature*, 521 (7553), p. 436.

[67] Elizabeth Leon, Olfa Nasraoui, and Jonatan Gomez, "Anomaly detection based on unsupervised niche clustering with application to network intrusion detection", in: *Proceedings of the 2004 congress on evolutionary computation (IEEE Cat. No. 04TH8753)*, vol. 1, IEEE, 2004, pp. 502–508.

[68] 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.

[69] Yuan Liu, Xiaofeng Wang, and Kaiyu Liu (2014), "Network anomaly detection system with optimized DS evidence theory", *The Scientific World Journal*, 2014.

[70] Chunlin Lu et al. (2016), "A Hybrid NIDS Model Using Artificial Neural Network and DS Evidence", *International Journal of Digital Crime and Forensics (IJDCF)*, 8 (1), pp. 37–50.

[71] Nemanja Maček and Milan Milosavljević (2014), "Reducing U2R and R2l category false negative rates with support vector machines", *Serbian Journal of Electrical Engineering*, 11 (1), pp. 175–188.

[72] Harshada C Mandhare and SR Idate, "A comparative study of cluster based outlier detection, distance based outlier detection and density based outlier detection techniques", in: *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)*, IEEE, 2017, pp. 931–935.

[73] Ahmed Mattar and Marek Z Reformat, "Detecting Anomalous Network Traffic Using Evidence Theory", in: *Advances in Fuzzy Logic and Technology 2017*, Springer, 2017, pp. 493–504.

[74] Yisroel Mirsky et al. (2018), "Kitsune: an ensemble of autoencoders for online network intrusion detection", *arXiv arXiv:1802.09089*.

[75] Nour Moustafa and Jill Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)", in: *2015 military communications and information systems conference (MilCIS)*, IEEE, 2015, pp. 1–6.

[76] Nour Moustafa and Jill Slay (2016), "The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set", *Information Security Journal: A Global Perspective*, 25 (1-3), pp. 18–31.

[77] Mary M Moya, Mark W Koch, and Larry D Hostetler (1993), "One-class classifier networks for target recognition applications", *NASA STI/Recon Technical Report N*, 93.

[78] Maya Nayak and Prasannajit Dash (2014), "Distance-based and Density-based Algorithm for Outlier Detection on Time Series Data", *Applied Science and Advanced Materials International*, p. 139.

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

[80] Atilla Özgür and Hamit Erdem (2016), "A review of KDD99 dataset usage in intrusion detection and machine learning between 2010 and 2015", *PeerJ Preprints*, 4, e1954v1.

[81] Leonid Portnoy (2000), "Intrusion detection with unlabeled data using clustering".

[82] K Saleem Malik Raja and K Jeya Kumar, "Diversified intrusion detection using Various Detection methodologies with sensor fusion", in: *2014 International Conference on Computation of Power, Energy, Information and Communication (ICCPEIC)*, IEEE, 2014, pp. 442–448.

[83] Deepthi Rajashekar, A Nur Zincir-Heywood, and Malcolm I Heywood, "Smart phone user behaviour characterization based on autoencoders and self organizing maps", in: *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, IEEE, 2016, pp. 319–326.

[84] Douglas A Reynolds (2009), "Gaussian Mixture Models.", *Encyclopedia of biometrics*, 741.

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

[86] Lukas Ruff et al., "Deep one-class classification", in: *International Conference on Machine Learning*, 2018, pp. 4393–4402.

[87] Mayu Sakurada and Takehisa Yairi, "Anomaly detection using autoencoders with nonlinear dimensionality reduction", in: *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis*, ACM, 2014, p. 4.

[88] Bernhard Schölkopf et al. (2001), "Estimating the support of a high-dimensional distribution", *Neural computation*, 13 (7), pp. 1443–1471.

[89] Bernhard Schölkopf et al. (2001), "Estimating the support of a high-dimensional distribution", *Neural computation*, 13 (7), pp. 1443–1471.

[90] Glenn Shafer (1976), *A mathematical theory of evidence*, vol. 42, Princeton university press.

[91] Kamran Shafi and Hussein A Abbass (2013), "Evaluation of an adaptive genetic-based signature extraction system for network intrusion detection", *Pattern Analysis and Applications*, 16 (4), pp. 549–566.

[92] Vrushank Shah, Akshai K Aggarwal, and Nirbhay Chaubey (2017), "Performance improvement of intrusion detection with fusion of multiple sensors", *Complex & Intelligent Systems*, 3 (1), pp. 33–39.

[93] Christos Siaterlis and Basil Maglaris, "Towards multisensor data fusion for DoS detection", in: *Proceedings of the 2004 ACM symposium on Applied computing*, ACM, 2004, pp. 439–446.

[94] Danaipat Sodkomkham et al. (2016), "Kernel density compression for real-time Bayesian encoding/decoding of unsorted hippocampal spikes", *Knowledge-Based Systems*, 94, pp. 1–12.

[95] 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.

[96] John A Swets (2014), *Signal detection theory and ROC analysis in psychology and diagnostics: Collected papers*, Psychology Press.

[97] Mahbod Tavallaee et al., "A detailed analysis of the KDD CUP 99 data set", in: *2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, IEEE, 2009, pp. 1–6.

[98] David MJ Tax and Robert PW Duin (2004), "Support vector data description", *Machine learning*, 54 (1), pp. 45–66.

[99] Marcio Andrey Teixeira et al. (2018), "SCADA system testbed for cybersecurity research using machine learning approach", *Future Internet*, 10 (8), p. 76.

[100] Nga Nguyen Thi, Van Loi Cao, and Nhien-An Le-Khac, "One-class collective anomaly detection based on lstm-rnns", in: *Transactions on Large-Scale Data-and Knowledge-Centered Systems XXXVI*, Springer, 2017, pp. 73–85.

[101] Ciza Thomas and N Balakrishnan, "Mathematical analysis of sensor fusion for intrusion detection systems", in: *2009 First International Communication Systems and Networks and Workshops*, IEEE, 2009, pp. 1–10.

[102] Ciza Thomas and N Balakrishnan (2009), "Improvement in intrusion detection with advances in sensor fusion", *IEEE Transactions on Information Forensics and Security*, 4 (3), pp. 542–551.

[103] Ciza Thomas and Balakrishnan Narayanaswamy (2010), "Mathematical basis of sensor fusion in intrusion detection systems", *Chapter 10 of Sensor Fusion and Its Applications*, pp. 225–250.

[104] Junfeng Tian, Weidong Zhao, and Ruizhong Du, "DS evidence theory and its data fusion application in intrusion detection", in: *International Conference on Computational and Information Science*, Springer, 2005, pp. 244–251.

[105] An Trung Tran (2017), "Network anomaly detection", *Future Internet (FI) and Innovative Internet Technologies and Mobile Communication (IITM) Focal Topic: Advanced Persistent Threats*, 55.

[106] Muhammad Usama et al. (2019), "Unsupervised machine learning for networking: Techniques, applications and research challenges", *IEEE Access*, 7, pp. 65579–65615.

[107] Kalyan Veeramachaneni et al., "AIˆ 2: training a big data machine to defend", in: *2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE In-*

ternational Conference on Intelligent Data and Security (IDS), IEEE, 2016, pp. 49–54.

[108] Kim Verbert, R Babuška, and Bart De Schutter (2017), "Bayesian and Dempster–Shafer reasoning for knowledge-based fault diagnosis–A comparative study", *Engineering Applications of Artificial Intelligence*, 60, pp. 136–150.

[109] Pascal Vincent et al., "Extracting and composing robust features with denoising autoencoders", in: *Proceedings of the 25th international conference on Machine learning*, ACM, 2008, pp. 1096–1103.

[110] Ly Vu et al., "Learning Latent Distribution for Distinguishing Network Traffic in Intrusion Detection System", in: *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, IEEE, 2019, pp. 1–6.

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

[112] Niklaus Wirth (1986), "Algorithms and data structures".

[113] Dit-Yan Yeung and Calvin Chow, "Parzen-window network intrusion detectors", in: *Object recognition supported by user interaction for service robots*, vol. 4, IEEE, 2002, pp. 385–388.

[114] Lotfi A Zadeh (1986), "A simple view of the Dempster-Shafer theory of evidence and its implication for the rule of combination", *AI magazine*, 7 (2), pp. 85–85.

[115] Matthew D Zeiler (2012), "Adadelta: an adaptive learning rate method", *arXiv arXiv:1212.5701*.

[116] Jiong Zhang, Mohammad Zulkernine, and Anwar Haque (2008), "Random-Forests-Based Network Intrusion Detection Systems", *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38, pp. 649–659.