A STUDY ON APPLYING SOME DEEP LEARNING ALGORITHMS FOR EARLY NETWORK INTRUSION DETECTION

Ho Thi Tuyen, Le Hoang Hiep*
TNU - University of Information and Communication Technology

ARTICLE INFO

Received: 08/3/2023
Revised: 21/4/2023
Published: 31/8/2023

KEYWORDS

Attack detection
Cyber attack
Network security
Deep learning
Network intrusion

ABSTRACT

This paper proposes and builds a model to evaluate the effectiveness of Deep Learning algorithms including Recurrent Neural Network (RNN), Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU), thereby determining the reliability of each dataset in building a network instrusion detection model. Because the models have similar structures, the evaluation will ensure objectivity. The results show that the algorithms applied on CICIDS2017 give a higher accuracy rate than the CSE-CICIDS2018 and the GRU model gives the best results. The study also shows that Deep Learning algorithms built on RNNs are relatively effective when it comes to detecting network attacks better than basic Machine Learning algorithms, which are capable of detecting a number of hidden features. both the CICIDS2017 and CSE-CICIDS2018 datasets are more reliable than the older ones.

NGHIÊN CỨU ỨNG DỤNG MỘT SỐ THUẬT TOÁN HỌC SÂU CHO BÀI TOÁN PHÁT HIỆN SỚM XÂM NHẬP BẤT THƯỜNG TRONG MẠNG

Hồ Thị Tuyến, Lê Hoàng Hiệp*
Trường Đại học Công nghệ thông tin và Truyền thông – ĐH Thái Nguyên

THÔNG TIN BÀI BÁO

Ngày nhận bài: 08/3/2023
Ngày hoàn thiện: 21/4/2023
Ngày đăng: 31/8/2023

TỪ KHÓA

Phát hiện tấn công
Tấn công mạng
An ninh mạng
Học sâu
Xâm nhập mạng

TÓM TẮT

Bài báo này tập trung nghiên cứu đề xuất xây dựng mô hình đánh giá hiệu quả của các thuật toán Deep Learning gồm Recurrent Neural Network (RNN), Long Short Term Memory (LSTM) và Gated Recurrent Unit (GRU), từ đó biết được mức độ tin cậy của từng bộ dữ liệu trong việc xây dựng mô hình phát hiện bất thường mạng. Do các mô hình đều có cấu trúc tương đồng nhau, vì vậy việc đánh giá sẽ đảm bảo tính khách quan. Hơn nữa, thông qua việc đánh giá hai bộ dữ liệu CICIDS2017 và CSE-CICIDS2018 kết quả cho thấy các thuật toán áp dụng trên CICIDS2017 cho tỉ lệ chính xác lên tới 98,96%, cao hơn so với bộ CSE-CICIDS2018 chỉ khoảng 89-91% và mô hình GRU cho kết quả tốt nhất (Accuracy trên CICIDS2017 là 98,73% và trên bộ CSE-CICIDS2018 là 91,76%). Nghiên cứu cũng cho thấy các thuật toán Deep Learning xây dựng dựa trên mạng RNN đều tỏ ra tương đối hiệu quả khi cho kết quả phát hiện tấn công mạng tốt hơn so với các thuật toán Machine Learning cơ bản, có khả năng phát hiện một số đặc trưng ẩn; cả hai bộ dữ liệu đều đáng tin cậy hơn so với những bộ dữ liệu đã cũ trước đây.

DOI: https://doi.org/10.34238/tnu-jst.7494

* Corresponding author. *Email: lhhiep@ictu.edu.vn*

1. Giới thiệu

Hệ thống phát hiện xâm nhập dựa trên sự bất thường (anomaly-based intrusion detection systems) là hệ thống giúp người dùng có thể ngăn chặn sớm các tác hại với hệ thống mạng của tổ chức. Đã có nhiều nghiên cứu về phương pháp tốt nhất cho hệ thống Intrusion Detection Systems (IDS), trong đó phương pháp học sâu (Deep Learning) là một trong những phương pháp được đánh giá cao [1], [2]. Deep Learning giúp người dùng có thể phát hiện các bất thường mạng với hiệu quả phát hiện ngày càng được nâng cao hơn so với Machine Learning [3], [4]. Các công bố liên quan tới nghiên cứu này gần đây đều xoay quanh một số thuật toán bao gồm Autoencoder, DNN, CNN, RNN/LSTM áp dụng vào bài toán phát hiện xâm nhập bất thường trong mạng [5], [6]. Các nghiên cứu này sử dụng một số kỹ thuật khác nhau (Mạng DNN với 3 lớp ẩn; Mạng Auto-Encoder bất đối xứng sử dụng cho học không giám sát; Mạng RNN và so sánh với các thuật toán học máy;…) với bộ dữ liệu khác nhau (NSL-KDD; KDD'99 NSL-KDD; CICIDS2017;…) trên các dạng tấn công mạng cũng khác nhau (DoS, R2L, U2R, Probe; Malicious file (shell code); Port Scan;…) nhằm cải thiện hoặc nâng cao kết quả phát hiện sự xâm nhập bất thường sớm trong mạng một cách hiệu quả [7], [8]. Có thể thấy các mô hình sử dụng thuật toán RNN và biến thể của nó như LSTM đều cho hiệu quả cao trong việc xây dựng hệ thống IDS Anomaly-based [9], [10]. Vì vậy trong nghiên cứu này sẽ đánh giá hiệu quả của mô hình RNN truyền thống cùng với 2 biến thể thường gặp của nó là LSTM và GRU cho bài toán phát hiện bất thường mạng.

2. Cơ sở nghiên cứu và thực hiện

2.1. Bộ dữ liệu CICIDS2017 và CSE-CICIDS2018

Một trong những thành phần quan trọng nhất cấu thành nên hệ thống IDS chính là hệ cơ sở tri thức. Để xây dựng nên hệ cơ sở tri thức, người dùng cần huấn luyện trước hệ thống với một bộ dữ liệu phản ánh cả hoạt động mạng bình thường và bất thường. Tuy nhiên vấn đề gặp phải trong suốt những năm gần đây đó là số bộ dữ liệu tốt còn hạn chế, cho kết quả chỉ tốt trên tập huấn luyện chứ không tốt trên thực tế. Khi quy mô của các cuộc tấn công ngày càng mở rộng, có nhiều dạng tấn công mới xuất hiện, những hệ thống IDS xây dựng dựa trên các bộ dữ liệu cũ như DARPA98, KDD'99, ISC2012, DEFCON,... ngày càng tỏ ra kém hiệu quả. Chính vì thế, nhu cầu xây dựng một bộ dữ liệu tốt cho hệ thống IDS luôn là nhu cầu mà các nhà nghiên cứu và phát triển mong muốn. Cả hai bộ dữ liệu CICIDS2017 và CSE-CICIDS2018 đều được xây dựng dựa trên công cụ CICFlowMeter – một công cụ cho phép trích xuất, thống kê các đặc trưng mạng từ file lưu lượng mạng (feature extraction). Từng mẫu dữ liệu trong tập đều được gắn nhãn đầy đủ, với hơn 80 đặc trưng mạng được thống kê. Bộ dữ liệu được lưu dưới dạng file "*.csv*" và được chia sẻ công khai.

2.2. Vai trò của một số thuật toán học máy

Qua cơ sở lý thuyết về các kiến trúc Deep Learning đã có, có thể thấy mỗi kiến trúc đều có ưu và nhược điểm riêng như [3] – [5]:

- ***Với Neural Network (ANN):*** Ưu điểm: mạng ANN có thể giải quyết các bài toán phức tạp, hay nói cách khác là tìm ra mối tương quan giữa input và output, thông qua việc học các trọng số (weight) và sử dụng activation function. Nhược điểm: Ma trận trọng số của ANN có kích thước lớn do cấu trúc Fully-connected. Ngoài ra mạng ANN gặp vấn đề Vanishing gradient/exploding gradient khi số lớp hidden layer lớn.

- ***Với RNN:*** Ưu điểm: mỗi unit trong mạng RNN không chỉ tiếp nhận thông tin từ input nó nhận được, mà còn tiếp nhận thông tin từ unit phía trước nó. Nhờ vậy output của mạng RNN mang độ liên kết thông tin hơn so với mạng ANN. Ngoài ra, số lượng tham số cần học là ít hơn. Nhược điểm: mạng RNN gặp vấn đề vanishing gradient/exploding gradient khi số lượng unit lớn.

- ***Với LSTM:*** Ưu điểm: do có kiến trúc tương tự như mạng RNN nên LSTM có các ưu điểm giống như RNN, đồng thời khắc phục được hiện tượng Vanishing gradient/Exploding gradient do

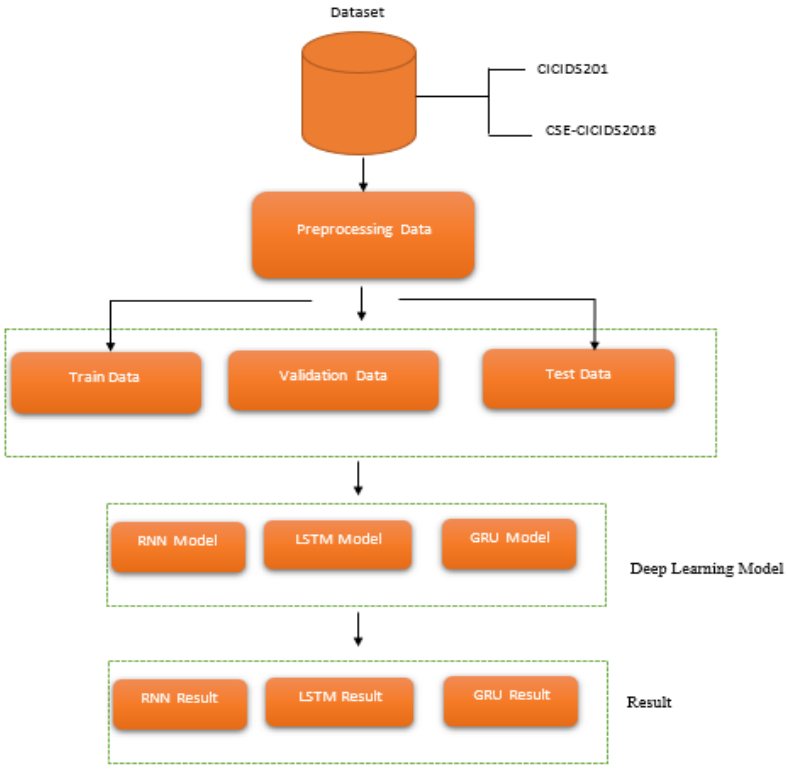
sử dụng trạng thái tế bào (Cell state). Nhược điểm: cấu trúc mỗi unit trong mạng LSTM phức tạp hơn vì sử dụng các cổng (gate). Số lượng cổng ở mỗi unit lớn nên việc training tốn nhiều thời gian hơn.

- ***Với GRU:*** Ưu điểm: tương tự như mạng LSTM, tuy nhiên số lượng cổng ít hơn nên tăng tốc độ training, trong khi vẫn giữ được các tính chất của mạng RNN. Nhược điểm: việc quản lý thông tin đến unit của mạng GRU không chặt chẽ bằng mạng LSTM do sử dụng ít cổng hơn.

Mặc dù lý thuyết về các mạng Deep Learning cho người dùng thấy được ưu nhược điểm của chúng, tuy nhiên người dùng cần áp dụng thực tế với bộ dữ liệu đề xuất CICIDS2017 và CSE-CICIDS2018 để có thể kiểm chứng hiệu quả hoạt động của mỗi thuật toán. Có thể thuật toán này hoạt động tốt trên một bộ dữ liệu, nhưng với bộ dữ liệu khác thuật toán này không còn hiệu quả nữa. Không chỉ vậy, người dùng cần hiệu chỉnh các tham số mô hình sao cho mô hình đạt hiệu quả tốt nhất.

3. Triển khai và đánh giá mô hình

Trong nghiên cứu này sẽ tập trung xây dựng một hệ thống phát hiện bất thường mạng dựa trên các thuật toán Deep Learning là ***RNN, LSTM và GRU*** [6] – [9]. Sơ đồ tổng quan của hệ thống như trong hình 1.



Hình 1. *Mô hình tổng quan hệ thống*

Hai bộ dữ liệu *CICIDS2017* và *CSE-CICIDS2018* sẽ được tiền xử lý để có thể đáp ứng yêu cầu huấn luyện, cũng như cải thiện độ chính xác của mô hình về sau. Đây là bước hết sức quan trọng, vì dữ liệu càng có độ chi tiết cao thì kết quả mô hình đem lại càng tốt. Sau đó dữ liệu sẽ được tách thành 3 tập Train, Test và Validation Data người dùng. Train Data người dùng là tập dữ liệu sử dụng cho mục đích huấn luyện, chiếm tỉ lệ 80% so với tổng cả bộ dữ liệu. Test Data người dùng là tập kiểm thử kết quả của mô hình, Validation Data người dùng là tập sử dụng vào mục đích giám sát quá trình huấn luyện. Hai tập Test và Validation chiếm tỉ lệ 10%. Sau khi dữ

liệu đã tách thành 3 tập trên, chúng được sử dụng để huấn luyện và đánh giá các mô hình RNN, LSTM và GRU.

3.1. Cấu trúc mô hình Deep Learning sử dụng

Các mô hình đều được xây dựng dựa trên kiến trúc của Neural Network [10]-[20]. Sau khi bộ dữ liệu CICIDS2017/CSE-CICIDS2018 đã được tiền xử lý, chúng được đưa qua lớp Reshape, sau đó đến lớp SimpleRNN/LSTM/GRU và đi đến các lớp Dense hay tên gọi khác là Fully-connected layer. Ngoài ra ở sau mỗi lớp Dense là một lớp Dropout Layer, thực hiện loại bỏ ngẫu nhiên tỉ lệ units ở lớp trước đó. Cuối cùng output của các mô hình là dự đoán mẫu lưu lượng thuộc lớp tấn công nào (từ 0 đến 14). Sơ đồ các mô hình như trong hình 2.

Hình 2. *Các mô hình RNN, LSTM và GRU*

Người dùng cần đưa dữ liệu gốc qua lớp Reshape Layer vì lớp RNN Layer/LSTM Layer/GRU Layer yêu cầu dữ liệu phải có dạng 3 chiều [batch, timesteps, feature]. Trong đó: *batch*: batch_size (số mẫu dữ liệu đưa vào tại một thời điểm); *timesteps*: số bước nhớ của mô hình; *features*: số đặc trưng của dữ liệu. Trước khi qua Reshape Layer, người dùng quy định dữ liệu sẽ đi vào theo từng mẫu một. Mỗi mẫu dữ liệu có 70 đặc trưng, vì vậy input_shape = (70). Vì vậy ở Reshape Layer người dùng cần dữ liệu vẫn đảm bảo có shape tương đương với input_shape (input_shape = timestep*features). Để dễ hình dung, người dùng đặt features = 70, timestep = 1. Sau khi qua lớp Reshape Layer, output sẽ được truyền vào lớp RNN/LSTM/GRU Layer, lớp này có chức năng phân tích các đặc trưng của dữ liệu. Output của lớp này có dạng 2 chiều (batch_size, units) với units là số unit có trong lớp. Ở các mô hình trên, output có dạng (None, 128). Hai lớp Dense Layer tiếp theo (512 units và 64 units) tiếp tục phân tích các đặc trưng dữ liệu. Ngoài ra sau mỗi Dense Layer có sử dụng một lớp Dropout Layer. Lớp Dropout Layer thực hiện loại bỏ ngẫu nhiên *k* % số unit ở lớp Dense Layer phía trước, mục đích của việc loại bỏ này nhằm tránh hiện tượng overfitting xảy ra trong quá trình huấn luyện. Lớp Dense Layer cuối cùng có 15 units, có nhiệm vụ biến đổi đầu vào dạng logits thành softmax (dạng xác suất). Với output dạng softmax, người dùng có thể sử dụng để tính hàm Loss cross-entropy áp dụng cho bài toán

phân loại (classification). Tổng số tham số ở các mô hình là: RNN: 125327 tham số; LSTM: 201743 tham số; GRU: 176655 tham số. Cuối cùng người dùng sử dụng Keras, áp dụng lần lượt các thuật toán với các hàm tối ưu, callback và tham số của chúng.

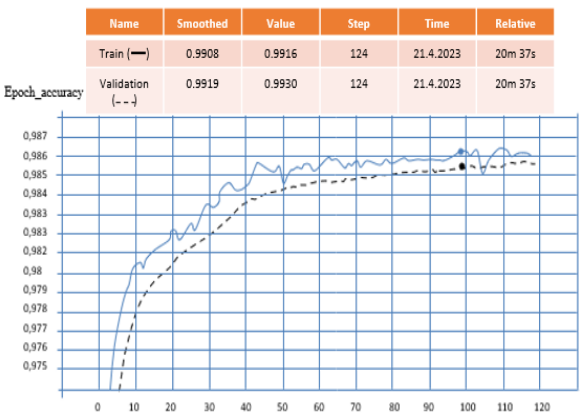
3.2. Đánh giá mô hình

3.2.1. Đánh giá theo Accuracy

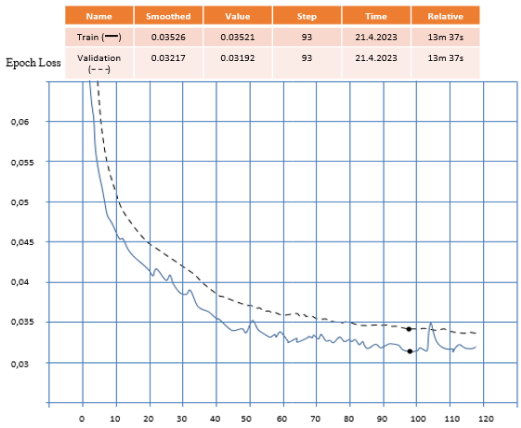
Nghiên cứu sử dụng công cụ Tensorboard để giám sát các mô hình trong quá trình huấn luyện chúng. Hai đại lượng được giám sát là Accuracy (độ chính xác) và Loss (mức độ sai lệch của dự đoán). Kết quả huấn luyện mô hình trên bộ dữ liệu CICIDS2017 được thể hiện như phần dưới đây. Độ mượt (Smoothing) của các biểu đồ là 0,6.

- Quá trình huấn luyện mô hình sử dụng thuật toán RNN truyền thống cho kết quả như biểu đồ hình 3 và hình 4.

Mô hình đạt kết quả tốt nhất tại epoch thứ 124 với Accuracy 99,51% và Loss 1,37% trên tập Validation. Như vậy có thể thấy mô hình LSTM tốt hơn so với RNN. Thời gian huấn luyện mô hình là 21 phút 58 giây.



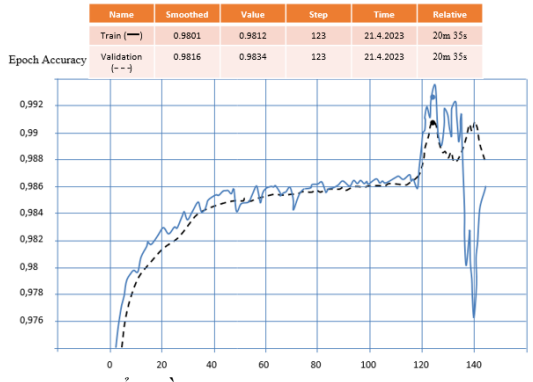
Hình 3. *Biểu đồ Accuracy – RNN – CICIDS2017 theo số Step*



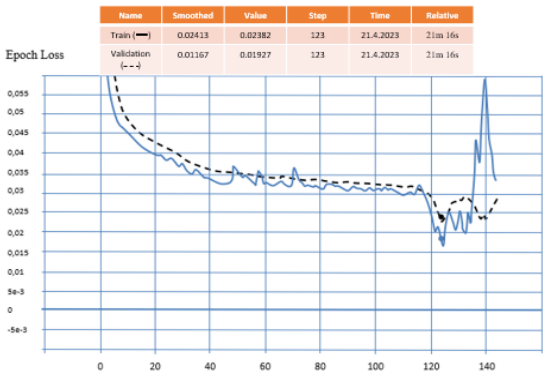
Hình 4. *Biểu đồ Loss - RNN – CICIDS2017 theo số Step*

Người dùng nhận thấy mô hình đạt kết quả tốt nhất tại epoch thứ 98, với Accuracy 98,63% và Loss 3,1% trên tập Validation. Thời gian huấn huyện mô hình là 15 phút 35 giây.

- Quá trình huấn luyện mô hình sử dụng thuật toán LSTM cho kết quả như biểu đồ hình 5 và hình 6.

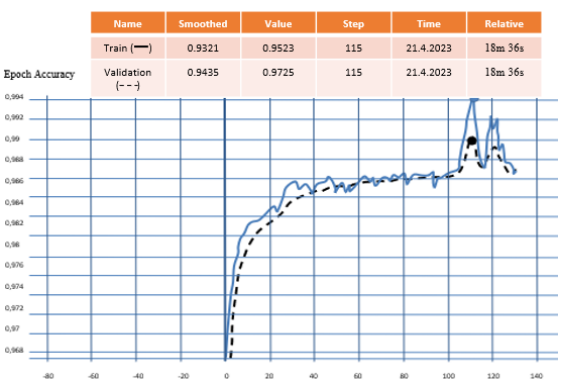


Hình 5. *Biểu đồ Accuracy - LSTM – CICIDS2017 theo số Step*

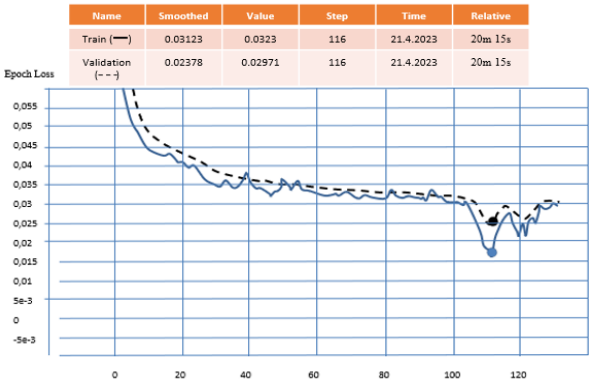


Hình 6. *Biểu đồ Loss – LSTM – CICIDS2017 theo số Step*

- Quá trình huấn luyện mô hình sử dụng thuật toán GRU cho kết quả như biểu đồ hình 7 và hình 8.



Hình 7. *Biểu đồ Accuracy – GRU – CICIDS2017 theo số Step*



Hình 8. *Biểu đồ Loss – GRU – CICIDS2017 theo số Step*

Mô hình đạt kết quả tốt nhất tại epoch thứ 112 với Accuracy 99,52% và Loss 1,46% trên tập Validation. Thời gian huấn luyện mô hình là 19 phút 48 giây, nhanh hơn so với mô hình LSTM. Với bộ dữ liệu CSE-CICIDS2018, quá trình huấn luyện được thể hiện như biểu đồ sau. Độ mượt (Smoothing) của các biểu đồ là 0,8.

Triển khai huấn luyện mô hình trên bộ dữ liệu ***CICIDS2018*** được tương tự các bước như với bộ dữ liệu *CICIDS2017*, kết quả cho thấy:

- Quá trình huấn luyện mô hình sử dụng thuật toán RNN truyền thống: Người dùng nhận thấy mô hình đạt kết quả tốt nhất tại epoch thứ 142, với Accuracy 93,8% và Loss 1,51% trên tập Validation. Thời gian huấn huyện mô hình là 27 phút 59 giây.

- Quá trình huấn luyện mô hình sử dụng thuật toán LSTM: Mô hình đạt kết quả tốt nhất tại epoch thứ 87 với Accuracy 93,83% và Loss 1,5% trên tập Validation. Như vậy có thể thấy mô hình LSTM tốt hơn so với RNN. Thời gian huấn luyện mô hình là 18 phút 52 giây.

- Quá trình huấn luyện mô hình sử dụng thuật toán GRU: Mô hình đạt kết quả tốt nhất tại epoch thứ 87 với Accuracy 93,82% và Loss 1,51% trên tập Validation. Thời gian huấn luyện mô hình là 18 phút 6 giây, nhanh hơn so với mô hình LSTM.

Kết quả đánh giá độ chính xác của các mô hình trên tập Test như trong Bảng 1.

Bảng 1. *Độ chính xác của mô hình trên tập Test*

Data người dùng	Model	Accuracy	Loss
CICIDS2017	RNN	0,9868	0,0316
	LSTM	0,9953	0,0158
	GRU	0,9955	0,0157
CSE-CICIDS2018	RNN	0,9385	0,1504
	LSTM	0,9387	0,1504
	GRU	0,9387	0,1505

Từ số liệu trong bảng 1 có thể thấy: Với bộ dữ liệu CICIDS2017, mô hình RNN cho độ chính xác trên tập Test thấp hơn LSTM và GRU (~1%), đồng thời Loss cao hơn (0,015). Với bộ dữ liệu CSE-CICIDS2018, mô hình RNN vẫn cho độ chính xác thấp hơn tuy nhiên không đáng kể (0,02%). Thuật toán LSTM và GRU cho độ chính xác và Loss trên tập Test gần tương đương nhau với cả hai bộ dữ liệu. Vì vậy người dùng cần đánh giá thêm bằng các phương pháp khác. Cả hai bộ dữ liệu CICIDS2017 và CSE-CICIDS2018 đều cho độ chính xác cao hơn. Các thuật toán học sâu liên quan đến RNN nhìn chung cho kết quả không tốt bằng thuật toán học máy KNN.

3.2.2. Đánh giá theo True/False Positive/Negative

Để đánh giá mô hình theo True/False Positive/Negative, người dùng coi các giá trị mang nhãn lành tính (BENIGN) được coi là giá trị Negative và tất cả các giá trị còn lại (giá trị mang nhãn tấn công) được coi là giá trị Positive. Kết quả các giá trị TPR, FPR, TNR, FNR với từng mô hình như trong Bảng 2.

Bảng 2. *Giá trị TPR, FPR, TNR, FNR của các mô hình*

Data người dùng	Model	TPR	FPR	TNR	FNR
CICIDS2017	RNN	0,979278	0,009542	0,990458	0,020722
	LSTM	0,995527	0,002714	0,997286	0,004473
	GRU	0,996041	0,002745	0,997255	0,003959
CSE-CICIDS2018	RNN	0,954895	0,007435	0,992564	0,045105
	LSTM	0,954483	0,007307	0,992693	0,045517
	GRU	0,955098	0,007791	0,992209	0,044902

Từ số liệu trong bảng 2 có thể thấy: Trên bộ dữ liệu CICIDS2017 mô hình RNN có tỉ lệ FPR cao hơn so với mô hình LSTM và GRU (~7%). Điều đó có nghĩa là tỉ lệ cảnh báo nhầm của mô hình RNN cao hơn nhiều. Đồng thời tỉ lệ FNR cũng lớn hơn mô hình LSTM và GRU (~2%), nghĩa là tỉ lệ không phát hiện tấn công của mô hình RNN lớn. Mô hình GRU cho kết quả tốt hơn mô hình LSTM khi tỉ lệ FNR thấp hơn (~0,5%). Với bộ dữ liệu CSE-CICIDS2018, cả ba mô hình đều cho kết quả tương đương nhau. Cả hai bộ dữ liệu CICIDS2017 và CSE-CICIDS2018 đều cho tỉ lệ cảnh báo nhầm thấp hơn. Từ Accuracy và FPR trên ta có thể khẳng định hai bộ dữ liệu này đáng tin cậy hơn bộ dữ liệu NSL-KDD, cũng như các bộ dữ liệu cũ.

3.2.3. Đánh giá theo Precision/Recall/F1-Score

Kết quả các giá trị Precision, Recall, F1-Score với từng mô hình như trong Bảng 3:

Bảng 3. *Giá trị Precision, Recall, F1-Score của các mô hình*

Data người dùng	Model	Precision	Recall	F1-Score
CICIDS2017	RNN	0,963426	0,979278	0,971287
	LSTM	0,989489	0,995527	0,992499
	GRU	0,989377	0,996041	0,992697
CSE-CICIDS2018	RNN	0,997143	0,954895	0,975561
	LSTM	0,997191	0,954483	0,97537
	GRU	0,997007	0,955098	0,975603

Từ số liệu trong bảng 3 ta thấy: Với bộ dữ liệu CICIDS2017, mô hình RNN có các kết quả Precision/Recall/F1-Score đều thấp hơn so với LSTM và GRU. Mô hình LSTM có Precision cao hơn GRU 0,01%, tuy nhiên có Recall thấp hơn 0,05%. F1-Score của mô hình LSTM thấp hơn GRU, cho thấy mô hình GRU đạt hiệu quả tốt nhất trong số 3 mô hình sử dụng. Với bộ dữ liệu CSE-CICIDS2018, các kết quả Precision/Recall/F1-Score của cả 3 mô hình xấp xỉ bằng nhau, tuy nhiên mô hình đạt F1-Score lớn nhất vẫn là mô hình GRU (97,56%). Giá trị Precision và Recall của các thuật toán LSTM và GRU nhìn chung không kém hơn nhiều so với thuật toán học máy KNN. Cùng với kết luận về Accuracy, ta có thể khẳng định các thuật toán học sâu LSTM và GRU vẫn tỏ ra rất hiệu quả trong bài toán phát hiện bất thường mạng.

4. Kết luận

Trong nghiên cứu này nhóm tác giả đã tiến hành phân tích, thực nghiệm và diễn giải ý tưởng cũng như mục đích và cách thức thực hiện. Kết quả đạt được bao gồm: Đánh giá hai bộ dữ liệu CICIDS2017 và CSE-CICIDS2018 bằng các thuật toán Deep Learning như RNN, LSTM và GRU, từ đó cho ta thấy mức độ tin cậy của từng bộ dữ liệu trong việc xây dựng mô hình phát hiện bất

thường mạng. Nhìn chung các thuật toán áp dụng trên bộ dữ liệu CICIDS2017 cho tỉ lệ Accuracy lên tới 98,96%, cao hơn so với bộ CSE-CICIDS2018 chỉ khoảng 89-91%. Xây dựng mô hình đánh giá hiệu quả của các thuật toán Deep Learning gồm RNN, LSTM và GRU. Các mô hình đều có cấu trúc tương đồng nhau, vì vậy việc đánh giá sẽ đảm bảo tính khách quan. Kết quả cho thấy mô hình GRU cho kết quả tốt nhất (Accuracy trên bộ CICIDS2017 là 98,73% và trên bộ CSE-CICIDS2018 là 91,76%, cùng các chỉ số đánh giá khác đã thống kê). Kết quả cũng là kênh thông tin hữu ích cho người dùng khi triển khai các giải pháp có sử dụng các thuật toán học sâu để nâng cao hiệu quả trong việc phát hiện sớm các xâm nhập mạng trái phép trong hệ thống của mình.

TÀI LIỆU THAM KHẢO/ REFERENCES

[1] S. Naseer and Y. Saleem, "Enhanced Network Intrusion Detection using Deep Convolutional Neural Networks," *KSII Transactions on internet and information systems*, vol. 12, no. 10, pp. 5159-5178, 2018.

[2] W. L. Al-Yaseen, Z. A. Othman, and M. Z. A. Nazri, "Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system," *Expert Syst. Appl.*, vol. 67, pp. 296–303, Jan. 2017.

[3] K. Alrawashdeh and C. Purdy, "Toward an Online Anomaly Intrusion Detection System Based on Deep Learning," *15th IEEE International Conference on Machine Learning and* Applications (ICMLA), USA, 2016, doi:10.1109/ICMLA.2016.0040

[4] V. K. Kanneganti, K. Swthi, and B. Brao, "A Novel Framework for NIDS through Fast kNN Classifier on CICIDS2017 Dataset," *International Journal of Recent Technology and Engineering*, vol. 8, no. 5, pp. 2277-3878, 2020.

[5] J. Kim, N. Shin, S. Y. Jo, and S. H. Kim, "Method of intrusion detection using deep neural network," in *Proc. of IEEE International Conference on Big Data and Smart Computing (BigComp)*, 2017, pp. 313–316.

[6] R. A. R. Ashfaq, X. Z. Wang, J. Z. Huang, H. Abbas, and Y. L. He, "Fuzziness based semi-supervised learning approach for intrusion detection system," *Information Sciences*, vol. 378, pp. 484–497, 2017, doi: 10.1016/j.ins.2016.04.019.

[7] E. Alhajjar, "Adversarial machine learning in Network Intrusion Detection Systems," *Expert Systems with Applications*, vol. 186, pp. 1-10, 2021.

[8] K. Swathi and B. B. Rao, "Impact of PDS Based kNN Classifiers on Kyoto Dataset," *International Journal of RoughSets and Data Analysis (IJRSDA)*, vol. 6, no. 2, pp. 61-72, 2019, doi: 10.4018/IJRSDA.2019040105.

[9] B. Rao, "A Fast KNN Based Intrusion Detection System for Cloud Environment," *Jour of Adv Research in Dynamical & Control Systems*, vol. 10, no. 7, pp. 1509 -1515, 2018.

[10] Y. Liao and R. V. Vemuri, "Use of K-Nearest Neighbor classifier for intrusion detection," *Computers & Security*, vol. 21, no. 5, pp. 439-448, 2002.

[11] F. S. D. L. Filho, A. M. B. Junior, G. V. Solar, and L. F. Silveira, "Smart Detection: An Online Approach for DoS/DDoS Attack Detection Using Machine Learning," *Security and Communication Networks*, vol. 2019, pp. 1-15, 2019.

[12] J. Long, "TR-IDS: Anomaly-Based Intrusion Detection through Text-Convolutional Neural Network and Random Forest," *Security and Communication Networks*, vol. 1, pp. 1-9, 2018.

[13] A. Maraj, "Testing of network security systems through DoS attacks," in *6th Mediterranean Conference on Embedded Computing (MECO)*, 2017, pp. 368-373, doi: 10.1109/MECO.2017.7977239.

[14] H. H. Le, "Studying a solution for early detection of DDoS attacks based on machine learning algorithms," *TNU Journal of Science and Technology*, vol. 227, no. 11, pp. 137 - 144, 2022.

[15] H. H. Le, "Study technique to limit bandwidth spending from DDoS attacks," *Yersin Journal of Science - Yersin University*, vol. 7, pp. 52-61, 2020.

[16] H. H. Le, "Improve network security system in Vietnam using reverse method," *TNU Journal of Science and Technology*, vol. 225, no. 09, pp. 125-133, 2020.

[17] H. H. Le, "Study to applying Blockchain technology for preventing of spam email," *TNU - Journal of Science and Technology*, vol. 208, no. 15, pp. 161-167, 2019.

[18] H. H. Le, "Combinning VLAN-Access List to enhance VLAN security efficient," *TNU Journal of Science and Technology*, vol. 181, no. 05, pp. 143-149, 2018.

[19] H. H. Le, "Network design of IPv6 safety based on analysis, feature assessment of IPv6 protocol," *TNU Journal of Science and Technology*, vol. 188, no. 12, pp. 85-91, 2018.

[20] H. H. Le, "Study to analyse, compare and evaluate the performance of Next General Firewalls: case of Palo Alto and Fortigate Firewall," *Vinh University Journal of Science (VUJS)*, vol. 51, no. 2A/2022, pp. 24-36, 2022.