Ứng dụng mô hình học sâu trong phát hiện tấn công trinh sát mạng

Nguyễn Thị Dung, Nguyễn Văn Quân, Nguyễn Việt Hùng

***Tóm tắt*— Ngày nay, cùng với sự phát triển nhanh chóng của Internet là thực trạng gia tăng các cuộc tấn công mạng cả về quy mô lẫn số lượng. Trong đó, tin tặc có thể sử dụng nhiều phương pháp tấn công khác nhau, nhưng tất cả thường diễn ra theo quy trình nhất định, bắt đầu từ bước trinh sát mạng. Chính vì vậy, để kịp thời phát hiện sớm các hành vi xâm nhập mạng trái phép, đặc biệt là từ giai đoạn trinh sát mạng, cần triển khai các giải pháp, hệ thống phát hiện xâm nhập ứng dụng với những kỹ thuật phát hiện tiên tiến. Trên thực tế, các hệ thống phát hiện xâm nhập mạng (Intrusion Detection System - IDS) thường dựa trên các dấu hiệu thông qua các luật đã được thiết lập trước. Kỹ thuật này còn nhiều hạn chế do không phát hiện được các cuộc tấn công mới hoặc biến thể của các cuộc tấn công đã biết. Nhằm khắc phục hạn chế này, nhiều kỹ thuật ứng dụng máy học đã được nghiên cứu và triển khai. Trong bài báo này, nhóm tác giả đề xuất hướng tiếp cận cải tiến mô hình mạng học sâu hai giai đoạn ứng dụng trong hệ thống phát hiện và phân loại các hình thức tấn công trinh sát mạng. Đề xuất sẽ được đánh giá, thử nghiệm với bộ dữ liệu tiêu chuẩn NSL-KDD, UNSW-NB15, CTU13.**

***Abstract*— In recent years, the number of new types of attacks has increased dramatically. Although there are many types of attack techniques, all of them are following the similar chain of attack, beginning with network reconnaissance phase. Therefore, network reconnaissance attack detection problem is important for every Intrusion Detection System (IDS). In fact, network intrusion detection systems are based on pre-defined rules so they are not able to detect new attacks or variants of known attacks. Meanwhile, hackers have developed many automated toolkits that allow subtle changes to the attack behavior sufficient for IDS to treat as a zero-day attack. To overcome this limitation, many machine learning models have been applied in IDS and implemented in a real network. In this paper, we propose a new approach that uses two stage AutoEncoder to detect network reconnaissance attacks. The proposed approach is evaluated on network security datasets: NSL-KDD, UNSW_NB15, four scenarios of the CTU13 datasets and compared to existing methods.**

Từ khóa— trinh sát mạng; phát hiện xâm nhập/bất thường; hệ thống phát hiện xâm nhập; học máy; mạng học sâu.

Keywords— network reconnaissance; anomaly detection; intrusion detection system; machine learning; deep learning.

I. GIỚI THIỆU

Trong các cuộc tấn công mạng, để khai thác vào hệ thống thông tin mục tiêu, trước hết tin tặc thường phải thăm dò về hệ thống đó và tìm kiếm những thông tin có giá trị. Thông qua các kỹ thuật trinh sát khác nhau, tin tặc sẽ cố gắng lấy thông tin về cấu hình và sơ đồ mạng, cách triển khai máy chủ của hệ thống mục tiêu, thông tin về cổng mở, dịch vụ cung cấp và các lỗ hổng tiềm ẩn trước khi thực hiện các hành vi khai thác xâm nhập vào mục tiêu sâu hơn. Kỹ thuật trinh sát một mạng máy tính có thể được phân loại thành ba nhóm chính, đó là trinh sát sử dụng các giao thức TCP, UDP và ICMP. Phản hồi nhận được từ phía nạn nhân cho biết các thông tin hữu ích để khởi động một cuộc tấn công, ví dụ như thông tin các cổng mở, các dịch vụ đang chạy, hệ điều hành máy chủ, lỗ hổng bảo mật,...

Các phương pháp tiếp cận phát hiện xâm nhập có thể được chia thành 4 loại như: phương pháp tiếp cận dựa trên thống kê, phương pháp tiếp cận dựa trên thuật toán, phương pháp tiếp cận dựa trên mô hình toán học, phương pháp tiếp cận dựa trên kinh nghiệm (heuristic).

Phương pháp tiếp cận thống kê: Các hệ thống dựa trên phương pháp này sẽ thu thập hành vi của người dùng và tạo một hồ sơ lưu lại hành vi của người dùng hợp pháp trong một

Bài báo được nhận ngày 05/4/2022. Bài báo được nhận xét bởi phản biện thứ nhất ngày 20/4/2022 và được chấp nhận đăng ngày 22/4/2022. Bài báo được nhận xét bởi phản biện thứ hai ngày 11/4/2022 và được chấp nhận đăng ngày 21/4/2022.

khoảng thời gian. Sau đó, các bài kiểm tra thống kê được áp dụng cho hành vi được quan sát để xác định với mức độ tin cậy cao xem đó có phải là hành vi của người dùng hợp pháp hay không. Khi có sự kiện diễn ra, hệ thống sẽ so sánh hoạt động đang diễn ra với hồ sơ được lưu, nếu điểm số bất thường cao hơn một ngưỡng nhất định thì hệ thống sẽ đưa ra cảnh báo.

Phương pháp tiếp cận dựa trên thuật toán: Sử dụng các quy trình từng bước để tính toán, xử lý dữ liệu và suy luận tự động (hệ thống có khả năng sử dụng tri thức đã lưu trữ để trả lời câu hỏi hay đưa ra kết luận hữu ích).

Phương pháp tiếp cận dựa trên mô hình toán học: Các phương pháp phát hiện quét mạng phân tán này sử dụng các mô hình toán học, máy trạng thái hữu hạn, các kỹ thuật về đại số và hình học khác để đạt được nhiệm vụ phát hiện tấn công mạng.

Các phương pháp tiếp cận theo phương pháp heuristic: Các phương pháp tiếp cận phát hiện trinh sát mạng phân tán này sử dụng phân tích dựa trên kinh nghiệm của các chuyên gia như trực quan, kinh nghiệm dựa trên bộ lọc, kinh nghiệm phân tích sự cố trước đó và các kỹ thuật tổng hợp khác.

Trong thời gian gần đây, nhiều nhóm nghiên cứu trên thế giới đã ứng dụng mô hình trí tuệ nhân tạo và tính toán thông minh vào IDS, bao gồm cả phát hiện tấn công trinh sát mạng. Trong bài báo này, nhóm tác giả sẽ đề xuất một phương pháp sử dụng mạng Autoencoder (AE) để giảm chiều dữ liệu đầu vào, sau đó áp dụng các thuật toán phân loại dữ liệu để phát hiện dấu hiệu của các cuộc tấn công trinh sát mạng. Với khả năng phân tích hiệu quả dữ liệu đa chiều lớn hơn, phương pháp đã thử nghiệm với bộ dữ liệu NSL_KDD, UNSW-NB15 và cho kết quả tốt hơn các phương pháp học máy trước đây.

Phần còn lại của bài báo được tổ chức như sau: trong Phần II, trình bày tổng quan về một số kỹ thuật tấn công trinh sát mạng và một số phương pháp phát hiện được nghiên cứu gần đây. Phần III trình bày về mạng học sâu AutoEncoder (AE), đề xuất mô hình học sâu ứng dụng trong phát hiện tấn công trinh sát mạng, phương pháp huấn luyện 2 giai đoạn là sử dụng mạng Autoencoder để giảm chiều dữ liệu, sau đó sử dụng các mô hình máy học phân loại dữ liệu. Phần IV là thực nghiệm và đánh giá kết quả thu được. Kết luận của nghiên cứu được trình bày trong Phần V.

II. CÁC NGHIÊN CỨU LIÊN QUAN

A. Kỹ thuật trinh sát mạng

Open Scan: Là kỹ thuật quét đơn giản [1]. Kỹ thuật này sử dụng giao thức TCP và cờ SYN để phát hiện các cổng TCP. Khi một cổng đóng, máy nạn nhân sẽ trả lời bằng cờ RST. Mặt khác, khi phát hiện một cổng đang mở, máy nạn nhân sẽ trả lời bằng cờ ACK. Một ưu điểm của kỹ thuật này là có thể quét một cách rất đơn giản mà không yêu cầu bất kỳ chức năng hoặc đặc quyền nào khác. Tuy nhiên, kỹ thuật đơn giản này sẽ dễ dàng bị phát hiện bởi tường lửa.

Half-Open Scan: Thường được gọi là quét TCP SYN, một phương pháp phổ biến để xác định cổng cho phép tin tặc thu thập thông tin về các cổng đang mở mà không cần hoàn thành quá trình bắt tay TCP. Vì kỹ thuật quét này không tạo ra một phiên kết nối TCP, nên không bị ghi log bởi các ứng dụng ở máy nạn nhân và ít ảnh hưởng đến máy chủ. Vì vậy nó không buộc ứng dụng khởi tạo hoặc phân bổ tài nguyên hệ thống. Tuy nhiên, do cần tạo các gói tin thô mới không hoàn toàn tuân theo bắt tay TCP nên quá trình kết nối nửa mở yêu cầu một số đặc quyền hệ thống nâng cao.

Stealth Scan: Các kỹ thuật trinh sát mạng ở trên chỉ sử dụng cờ SYN điển hình để kiểm tra các cổng đang mở. Do đó, chúng dễ dàng bị phát hiện và ghi lại bởi các IDS. Kỹ thuật Stealth Scan cố gắng tránh các thiết bị lọc bằng cách sử dụng một số bộ cờ khác với SYN để xuất hiện dưới dạng lưu lượng truy cập hợp pháp. Đó là SYN-ACK Scan, IDLE Scan, Fin, Xmas Tree and Null Scans, ACK Scan, Windows Scan và TCP Fragmentation Scan.

Sweep Scan: Kỹ thuật này không nhằm mục đích xác định các cổng đang hoạt động mà là xác định các thiết bị đang hoạt động. Đặc điểm của kỹ thuật này là thực hiện quét hàng loạt, mục đích là xác định trạng thái của càng nhiều máy càng tốt thay vì tập trung vào một máy

riêng lẻ. Sweep Scan hoạt động bằng cách tạo ra yêu cầu phản hồi của các máy trạm từ xa. Các kỹ thuật Sweep Scan là ICMP Echo Request Scan, ICMP Timestamp v Address Mask Scans và TCP SYN Scan.

Miscellaneous Scan: Là các kỹ thuật quét sử dụng các giao thức khác với các giao thức ở trên. Các kỹ thuật này bao gồm FTP bounce, UDP, giao thức IP và RPC scans.

Các kỹ thuật tấn công này thường được xây dựng trong các bộ dữ liệu an ninh mạng tiêu chuẩn dùng để đánh giá các mô hình phát hiện tấn công mạng, ví dụ như các bộ dữ liệu được sử dụng trong bài báo này.

B. Các phương pháp phát hiện trinh sát mạng

Trong phần này, nhóm tác giả sẽ trình bày một số kỹ thuật phát hiện trinh sát mạng đã được một số nhóm nghiên cứu đề xuất gần đây.

Tác giả M. Vidhya đã sử dụng kỹ thuật trích chọn đặc trưng và máy vectơ hỗ trợ (SVM) để phân loại các cuộc tấn công trinh sát mạng [3]. Tác giả đề xuất phương pháp trích xuất đặc trưng cho tập dữ liệu KDD99 bằng cách sử dụng thuật toán Consistency Sybset Evaluation và phương pháp Best First. Kết quả thử nghiệm cho thấy sử dụng SVM với tập dữ liệu đã được giảm chiều chính xác hơn với tập dữ liệu gốc. Các tác giả đạt hiệu suất khá cao là 99,9185% [3]. Tuy nhiên, với dữ liệu không cân bằng, các nhóm nghiên cứu khác thường sử dụng phép đo liên quan đến dữ liệu tấn công.

Nhóm tác giả Meijuan Gao cùng các cộng sự đã trình bày một IDS dựa trên SVM, sử dụng các thuật toán tiến hóa để tối ưu hóa các thông số SVM nhằm cải thiện độ chính xác trong quá trình phát hiện xâm nhập [4].

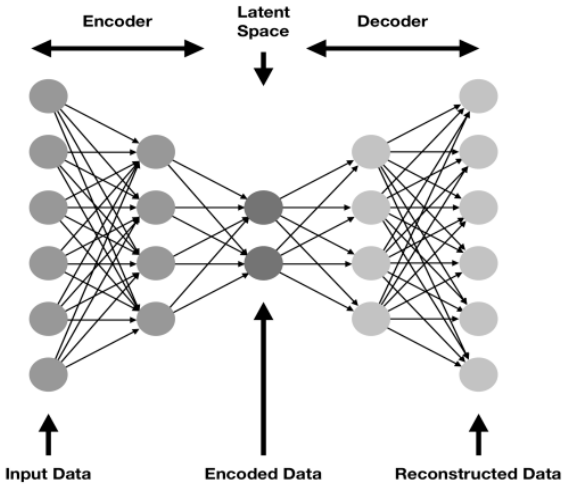
Giải pháp AOCD (An Adaptive Outlier Based Scope-Based Scalarized Scan Detection Approach) đã được Monowar H. Bhuyan cùng các cộng sự nghiên cứu và giới thiệu nhằm phát hiện sớm tấn công trinh sát mạng có độ chính xác cao khi thử nghiệm với bộ dữ liệu KDD99 [5]. Tác giả đã trình bày một giải pháp để chuyển đổi dữ liệu lưu lượng mạng thành một định dạng mà các bộ lọc và bộ phân loại có thể xử lý. Điều đáng chú ý là nhóm tác giả đã lựa chọn các mẫu ngẫu nhiên trong cơ sở dữ liệu, từ đó xác định tập các thuộc tính cho việc phát hiện phân cụm được sử dụng trong kỹ thuật phát hiện quét mạng trước kia. Nhóm nghiên cứu đã thử nghiệm trên dữ liệu thực thu thập được từ Nhóm tác giả Thang. N. M và Luong T. T [10] cũng sử dụng học máy trong phát hiện tấn công và trinh sát website với bộ 8 thuộc tính tự định nghĩa, cho kết quả khả quan. Hệ thống phát hiện xâm nhập của Đại học Tezpur của Ấn Độ, kết quả chứng minh giải pháp này có khả năng phát hiện các cuộc tấn công mạng với độ chính xác cao [5].

III. ỨNG DỤNG HỌC SÂU PHÁT HIỆN TẤN CÔNG TRINH SÁT MẠNG

A. Mạng học sâu Autoencoder

Trong thời gian gần đây, các mô hình học sâu đã được ứng dụng thành công trong nhiều lĩnh vực như nhận dạng hình ảnh, xử lý giọng nói, xử lý ngôn ngữ tự nhiên và xử lý dữ liệu lớn.

Autoencoder là một loại mạng nơ-ron nhiều lớp, được thiết kế với mục đích mã hóa dữ liệu đầu vào thành các biểu diễn tiềm ẩn, sau đó giải mã để cố gắng tái tạo lại đầu vào bằng cách sử dụng dữ liệu mã hóa vừa được tạo. Kiến trúc mạng Autoencoder được chia thành 3 phần: Bộ mã hóa (Encoder), Bộ giải mã (Decoder) và Không gian tiềm ẩn (Latent Space) hay còn được gọi là "nút cổ chai" (Bottleneck) như trong Hình 1.



Hình 1. Mô hình mạng Autoencoder

Để học các biểu diễn dữ liệu của đầu vào, mạng được huấn luyện bằng cách sử dụng dữ liệu không được giám sát (không được gán nhãn). Dữ liệu đầu vào sẽ được biến đổi (mã hóa) qua các lớp nơ-ron sang không gian có số chiều ít hơn (không gian tiềm ẩn). Các biểu diễn dữ liệu ở không gian tiềm ẩn này sẽ trải qua một quá trình giải mã, với mục tiêu tái tạo đầu vào.

Bộ mã hóa: Bao gồm một mạng nơ-ron truyền thẳng thông thường, được xây dựng để dự đoán biểu diễn tiềm ẩn của dữ liệu đầu vào. Trong đó f_θ là Bộ mã hóa và $X = \{x_1, x_2, ..., x_n\}$ là một tập dữ liệu. Bộ mã hóa f_θ nhằm ánh xạ đầu vào $x_i \in X$ thành một đại diện ẩn (latent representation) [7].

$$z_i = f_\theta(x_i) \quad (1)$$

Bộ giải mã: bao gồm mạng nơ-ron truyền thẳng nhưng kích thước của của các lớp tăng theo thứ tự ngược lại (đối xứng) với các lớp của Bộ mã hóa. Bộ giải mã g_ϕ nhằm mục đích ánh xạ biểu diễn tiềm ẩn z_i trở lại không gian đầu vào, do đó, việc tái tạo được tính theo Công thức 2 [7]:

$$\hat{x}_i = g_\phi(z_i) \quad (2)$$

Bộ mã hóa và Bộ giải mã được biểu diễn dưới dạng các hàm kích hoạt của các hàm tuyến tính liên quan đến trọng số và độ lệch [7] như sau:

$$f_\theta(x) = \psi_f(\mathrm{W}x + b) \quad (3)$$

$$g_\phi(x) = \psi_g(\mathrm{W}'z + b') \quad (4)$$

Trong đó θ = (W, b) và ϕ = (W ', b') lần lượt là (trọng số và độ lệch) cho Bộ mã hóa và Bộ giải mã. ψ_f và ψ_g là các hàm kích hoạt tương ứng.

Hàm mất mát tái tạo trên các mẫu huấn luyện [7] có thể được viết như sau:

$$\mathcal{L}_{AE}(\theta; \phi; x) = \frac{1}{n}\sum_{i=0}^{n} l(x_i, \hat{x}_i) \quad (5)$$

Trong đó $l(x_i, \hat{x}_i)$ là sự khác biệt giữa đầu vào x_i và sự tái cấu trúc của nó $\hat{x}_i$ với $\hat{x}_i = g_\phi(f_\theta(x_i))$; n là số lượng mẫu dữ liệu trong tập dữ liệu. Autocencoder học cách tối ưu hóa hàm mục tiêu trong (5) liên quan đến các tham số θ và ϕ bằng cách sử dụng phương pháp học lan truyền ngược.

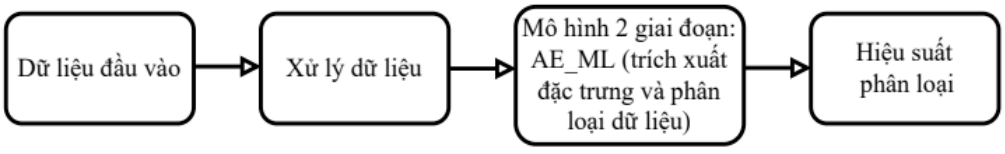
Mạng AE được ứng dụng nhiều trong khoa học dữ liệu như: giảm chiều kích thước, phát hiện bất thường, khử nhiễu hình ảnh, nén hình ảnh, tạo hình ảnh, trích xuất đặc trưng, hệ thống gợi ý.

B. Mô hình đề xuất

Nhóm tác giả đề xuất mô hình 2 giai đoạn ứng dụng trong phát hiện và phân loại các hình thức tấn công trinh sát mạng là sử dụng mạng AE kết hợp các thuật toán học máy truyền thống (AE_Decision Tree, AE_RandomForest, AE_kNN, AE_Naive Bayes, AE_MLP, AE_SVM) (Hình 2). Trong đó:

- Giai đoạn 1 là dùng mạng AE để huấn luyện dữ liệu không giám sát nhằm trích chọn đặc trưng.
- Giai đoạn 2 là học có giám sát, dữ liệu được lấy từ không gian tiềm ẩn của mạng AE làm đầu vào của thuật toán học máy để phát hiện và phân loại tấn công.

Ý tưởng của đề xuất là thay vì đưa tất cả số chiều của dữ liệu gốc vào các thuật toán phân lớp cổ điển thì sẽ dùng mạng AE để giảm chiều dữ liệu (trích chọn đặc trưng), sau đó đưa vào các thuật toán phân loại. Mục đích nhằm có thể cải thiện hiệu suất của các mô hình phân loại dữ liệu cổ điển.



Hình 2. Sơ đồ tổng quát của mô hình đề xuất

IV. KẾT QUẢ VÀ ĐÁNH GIÁ

A. Dữ liệu thực nghiệm

1. Tập dữ liệu NSL-KDD

NSL-KDD là bộ dữ liệu được sử dụng rộng rãi với các IDS hiện nay. NSL-KDD được cải tiến từ tập dữ liệu KDD99, giúp loại bỏ dữ liệu trùng lặp, dữ liệu dư thừa, được bổ sung thêm nhiều dữ liệu huấn luyện và dữ liệu thử nghiệm hơn. Tập dữ liệu gồm 2 phần, một để huấn luyện gồm 125.973 bản ghi (*KDDTrain+.csv*), một tập để kiểm thử gồm 22.544 bản ghi (*KDDTest+.csv*). Bộ dữ liệu có 41 thuộc tính như trong KDD99, được thu thập và gắn nhãn từ các cuộc tấn công thử nghiệm khác nhau. Có bốn kiểu tấn công trinh sát mạng là nmap, ipsweep, portsweep và satan được mô tả trong Bảng 1.

BẢNG 1. DỮ LIỆU TẤN CÔNG TRINH SÁT MẠNG TRONG NSL-KDD

Kiểu tấn công	**Dữ liệu huấn luyện**	**Dữ liệu kiểm thử**
IPSweep	3599	141
Nmap	1493	73
PortSweep	2931	157
Satan	3633	735
Tổng	11656	1106

2. Tập dữ liệu UNSW-NB15

Bộ dữ liệu KDD98, KDD99, NSL-KDD đã được sử dụng từ lâu nên nhiều kỹ thuật tấn công dùng để tạo dữ liệu đã lỗi thời, không cập nhật các kỹ thuật tấn công hiện có. Bộ dữ liệu UNSW-NB15 đã khắc phục được hạn chế của các bộ dữ liệu trước đó và đang được sử dụng rộng rãi trong các nghiên cứu phát hiện xâm nhập. Bộ dữ liệu UNSW-NB15 được xây dựng vào năm 2015 tại Phòng thí nghiệm Cyber Range của Đại học New South Wales.

Dữ liệu tấn công được tạo tự động từ hệ thống IXIA PerfectStorm với nhiều kỹ thuật tấn công mới. Các gói tin được bắt, xử lý trước và trích xuất thành 49 thuộc tính [8]. Có tổng cộng 175.341 bản ghi trong tập dữ liệu huấn luyện và 82.332 bản ghi trong tập dữ liệu thử nghiệm.

3. Bộ dữ liệu CTU13

Bộ dữ liệu CTU-13 được nghiên cứu bởi Đại học Kỹ thuật Cộng hòa Séc và được công bố năm 2011. Đây là bộ dữ liệu chứa thông tin bao gồm cả 3 loại lưu lượng, đó là lưu lượng Botnet, lưu lượng bình thường (normal traffic) và lưu lượng nền (background traffic) của hạ tầng dịch vụ mạng.

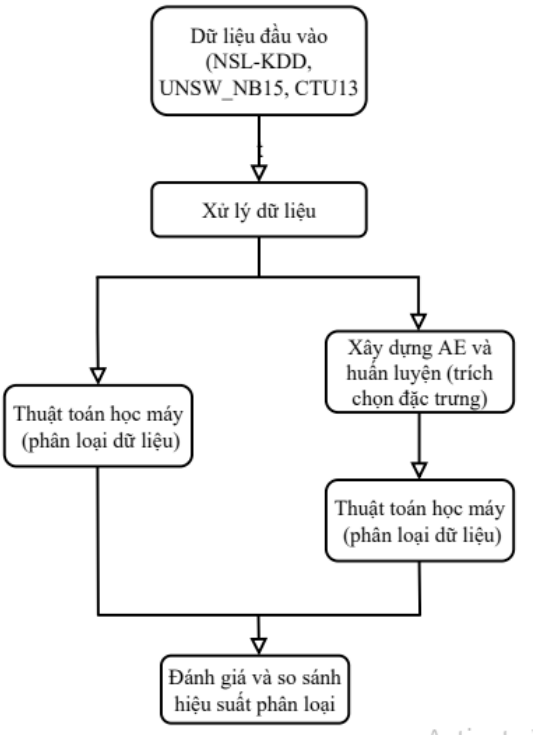
Các bộ dữ liệu con có số các thuộc tính khác nhau và được đánh tên theo ký kiệu từ CTU-13_01 đến CTU-13_13, trong đó các bộ dữ liệu đang được nghiên cứu nhiều để đưa ra đánh giá kết quả trong lĩnh vực Học máy, Học sâu là CTU13_08, CTU13_10, CTU13_12 và CTU13_13. Nghiên cứu của nhóm tác giả sẽ thử nghiệm trên 4 bộ dữ liệu này.

B. Thực nghiệm

Thực nghiệm được thực hiện với 2 phương pháp, một là phát hiện và phân loại tấn công khi sử dụng các mô hình học máy truyền thống; hai là thực nghiệm với mô hình đề xuất kết hợp sử dụng mạng AE để trích chọn đặc trưng của tập dữ liệu ban đầu, sau đó làm đầu vào cho các mô hình học máy để phát hiện và phân loại (Hình 3).

1. Xử lý dữ liệu

Các bộ dữ liệu đầu vào sẽ được tiền xử lý trên cả 2 tập, tập huấn luyện và tập kiểm thử. Các thuộc tính kiểu phân loại sẽ được xử lý thông qua thuật toán One-hot Encoding. Sau đó, dữ liệu được chuẩn hóa theo một trong số phương pháp chuẩn hóa (Scale) như *StandardScaler, MaxAbsScaler* được cung cấp trong thư viện sklearn.



Hình 3. Lưu đồ thực nghiệm của nghiên cứu

2. Xây dựng và huấn luyện AutoEncoder

Chương trình huấn luyện mạng Autoencoder được xây dựng bằng Python với thư viện Pytorch. Thử nghiệm được tiến hành trên PC Intel Xeon Core i5-3.6 GHz với 12GB RAM. Đối với từng bộ dữ liệu đầu vào (phụ thuộc véc tơ dữ liệu đầu vào có bao nhiêu giá trị), kiến trúc mạng AE được xây dựng với số tầng, số nút tương ứng khác nhau. Vectơ đầu vào của mạng có kích thước bằng với số lượng các thuộc tính của bộ dữ liệu được mô tả ở trên. Cấu hình cuối cùng của mô hình AE đã được chọn thông qua các quá trình thử nghiệm và Bộ mã hóa của AE có kiến trúc áp dụng với từng bộ dữ liệu như Bảng 2.

BẢNG 2. KIẾN TRÚC MẠNG AE VỚI CÁC DATASET

Bộ dữ liệu	Kiến trúc mạng AE (Bộ mã hóa _Encoder)				
	Đầu vào	Tầng ẩn 1	Tầng ẩn 2	Tầng ẩn 3	Không gian tiềm ẩn
NSL-KDD	122	85	49		12
UNSW	196	150	100	40	16
CTU13	40	30	15		7

Các tham số được đưa vào huấn luyện mạng như sau: leaning_rate = 0.01, batch_size = 100, display_step = 10, epoch = 200.

Trọng số của AE được khởi tạo bằng phương pháp khởi tạo Xavier [9] để tăng tốc quá trình hội tụ, thuật toán tối ưu hóa là Adadelta, hàm kích hoạt là hàm TANH.

3. Thuật toán học máy

Một số thuật toán được sử dụng để phân loại dữ liệu tấn công trinh sát mạng như thuật toán Decision Tree, Random Forest, Naive Bayes, K-Nearest Neighbor (k-NN), Multi-layer Perceptron (MLP), Support Vector Machine (SVM).

C. Kết quả và đánh giá

Kết quả của bộ phân loại thường được đánh giá thông qua ma trận nhầm lẫn và một số phép đo dựa trên ma trận này. Ma trận nhầm lẫn mô tả trong Bảng 3, trong đó Class 0 là bình thường, Class 1 là tấn công.

BẢNG 3. MA TRẬN NHẦM LẪN

		Lớp dự đoán	
		Class 0	**Class 1**
Lớp thực tế	**Class 0**	TN	FN
	Class 1	FP	TP

Các độ đo sau đây sẽ được sử dụng để đánh giá hiệu suất của các mô hình.

$$acc = \frac{TP + TN}{TP + FP + TN + FN} \quad (6)$$

$$Precision(\pi) = \frac{TP}{TP + FP} \quad (7)$$

$$\mathrm{Re}call(\rho) = \frac{TP}{TP + FN} \quad (8)$$

$$F1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \quad (9)$$

1. Bộ dữ liệu NSL-KDD

Đối với bộ dữ liệu NSL-KDD sẽ được thực nghiệm với hai bộ phân loại: Bộ thứ nhất là bộ phân loại nhị phân để xác định xem bản ghi lưu lượng mạng thu được là tấn công hay bình

thường. Bộ phân loại thứ hai được huấn luyện để phân loại bốn kiểu tấn công trinh sát mạng *(nmap, ipsweep, portsweep và satan)*.

Phân loại nhị phân

Bảng 4 mô tả ma trận nhầm lẫn thu được khi sử dụng thuật toán phân loại kNN và khi sử dụng mô hình đề xuất AE_kNN, để phân loại dữ liệu là tấn công hay bình thường.

Bảng 5 tổng hợp hiệu suất phân loại nhị phân đạt được khi sử dụng các thuật toán máy học truyền thống (ML) và hiệu suất phân loại khi sử dụng mô hình đề xuất (AE_ML).

Dựa trên kết quả thu được từ Bảng 5, cho thấy mô hình đề xuất đạt hiệu suất cao hơn với tỉ lệ 4/6 mô hình được thử nghiệm, trong đó các mô hình cho hiệu suất tốt hơn là khi kết hợp AE với các thuật toán phân loại là AE_Naive Bayes, AE_kNN, AE_MLP, AE_SVM.

Cụ thể: 02 mô hình AE_Naive Bayes và AE_kNN cho kết quả cao hơn trên cả 4 độ đo. Hai mô hình AE_MLP, AE_SVM có hiệu suất tốt hơn rõ rệt với các chỉ số (ở 3 độ đo) cao hơn trong khoảng 2,23% - 39,29%, trong khi chỉ có một độ đo ở mỗi mô hình có độ chênh lệch hiệu suất thấp hơn trong khoảng 2,89% - 3,19% so với thuật toán MLP và SVM.

BẢNG 4. MA TRẬN NHẦM LẪN - NSL-KDD

		Kết quả dự đoán của kNN		**Kết quả dự đoán của AE_kNN**	
		Attack	**Normal**	**Normal**	**Attack**
Dữ liệu thực	**Attack**	9426	**285**	9451	**260**
	Normal	**263**	843	**123**	983

BẢNG 5. KẾT QUẢ PHÁT HIỆN TẤN CÔNG VỚI BỘ DỮ LIỆU NSL-KDD

Phương pháp	**ML (%)**				**AE_ML (%)**			
	acc	*π*	*Recall*	*F1*	*acc*	*π*	*Recall*	*F1*
DT	**96.43**	**78.71**	**89.24**	**83.64**	90.90	53.77	78.66	63.87
RF	**96.86**	**81.45**	**89.69**	**85.37**	96.73	84.50	83.27	83.88
Naive Bayes	10.33	10.19	99.46	18.49	**93.60**	**61.54**	**99.82**	**76.14**
K_NN	94.93	74.73	76.22	75.47	**96.46**	**79.08**	**88.88**	**83.70**
MLP	77.78	31.51	**100**	47.92	**95.61**	**70.80**	97.11	**81.89**
SVM	94.40	**78.87**	61.75	69.27	**96.63**	75.68	**98.73**	**85.68**

Phân loại 4 kiểu tấn công trinh sát mạng

Kết quả từ Bảng 6 cho thấy đối với phân loại 4 kiểu tấn công trinh sát mạng, mô hình đề xuất cho hiệu quả phân loại cao hơn là AE_DT, AE_Naive Bayes, AE_SVM, AE_MLP với chỉ số của các độ đo acc, F1, recall, precision đa số là cao hơn so với thuật toán học máy thông thường.

2. Bộ dữ liệu UNSW-NB15

Đối với tập dữ liệu UNSW-NB15, vì các kiểu tấn công trinh sát mạng được gán nhãn giống nhau (nhãn 1) nên chỉ áp dụng phân loại nhị phân để xác định xem dữ liệu có phải là một cuộc tấn công hay không. Do đối với tập dữ liệu UNSW-NB15 kiểm thử là tập dữ liệu tương đối cân bằng giữa 2 lớp. Trong đó: Lớp normal (0) có 37.000 bản ghi (44.94%) và lớp attack (1) có 45332 bản ghi (55,06%), do đó có thể dùng độ đo *accuracy* để đánh giá và so sánh hiệu suất của các phương pháp được sử dụng (Hình 4).

Kết quả cho thấy, mô hình đề xuất có hiệu suất cao hơn ở hầu hết các thuật toán. Cụ thể, 5 mô hình có *accuracy* cao hơn là AE_DT, AE_RF, AE_Naive Bayes, AE_kNN, AE_SVM (Hình 4), trong đó 03 mô hình có kết quả cao hơn ở tất cả các độ đo là AE_DT, AE_RF, AE_kNN (Bảng 7).

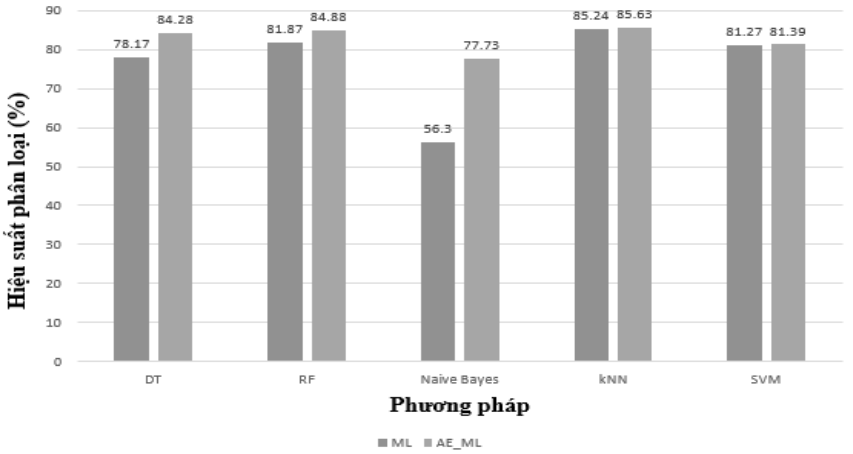
BẢNG 6. KẾT QUẢ PHÁT HIỆN 4 KIỂU TẤN CÔNG CỦA CÁC PHƯƠNG PHÁP

Phương pháp	**(%)**	**Kiểu tấn công**			
		Ipsweep	**Nmap**	**Portsweep**	**Satan**
DT	acc	93.50			
	π	2.44	0.00	**64.60**	78.76
	recall	0.71	0.00	**92.99**	64.08
	F1	1.10	0,00	**76.24**	70.67
AE_ DT	acc	**95.03**			
	π	**87.82**	**4.35**	32.96	**81.19**
	recall	**97.16**	**1.37**	56.05	**74.01**
	F1	**92.26**	**2.08**	41.51	**77.44**
Naive Bayes	acc	7.55			
	π	0.00	0.00	0.29	19.03
	recall	0.00	0.00	12.74	75.51
	F1	0.00	0.00	0.56	30.40
AE_ Naive Bayes	acc	**94.33**			
	π	**57.92**	**29.05**	**69.15**	**69.29**
	recall	**98.58**	**71.23**	**88.54**	**97.01**
	F1	**72.97**	**41.27**	**77.65**	**80.84**
SVM	acc	94.41			
	π	0.00	**100**	59.43	**98.62**
	recall	0.00	50.69	**92.36**	58.50
	F1	0.00	67.27	72.32	73.44

Phương pháp	(%)	Kiểu tấn công			
		Ipsweep	Nmap	Portsweep	Satan
AE_SVM	acc	**96.54**			
	π	**73.49**	64.49	**69.11**	79.47
	recall	**86.53**	**94.52**	84.08	**97.42**
	F1	**79.48**	**76.67**	**75.86**	**87.53**
kNN	acc	94.82			
	π	89.61	90.12	46.04	82.98
	recall	97.87	100	96.18	63.67
	F1	93.56	94.80	62.27	72.05
AE_kNN	acc	**96.49**			
	π	81.82	71.57	**62.45**	80.61
	recall	31.92	100	94.27	**96.74**
	F1	45.92	83.43	**96.74**	**87.94**
MLP	acc	84.51			
	π	**89.54**	12.12	12.12	51.47
	recall	**97.16**	21.92	**96.18**	**64.08**
	F1	**93.20**	15.61	21.52	57.09
AE_MLP	acc	**94.72**			
	π	50.00	**33.65**	**43.96**	**83.44**
	recall	17.02	**97.26**	83,44	57.58
	F1	25.40	**50.00**	**57,58**	**86.98**

BẢNG 7. KẾT QUẢ PHÁT HIỆN TẤN CÔNG VỚI BỘ DỮ LIỆU UNSW_NB15

Phương pháp	ML (%)			AE_ML (%)		
	π	Recall	F1	π	Recall	F1
DT	79.39	81.53	80.44	**79.86**	**95.54**	**87.00**
RF	75.35	99.67	85.82	**79.04**	**99.87**	**87.79**
K_NN	80.57	96.43	87.79	**80.83**	**96.86**	**88.13**
MLP	**76.69**	98.79	**86.35**	75.39	**99.35**	85.72
SVM	74.77	**99.61**	85.42	**74.92**	99.51	**85.49**



Hình 4. Biểu đồ so sánh acc của mô hình đề xuất UNSW_NB15

3. Bộ dữ liệu CTU_13

Đối với tập dữ liệu CTU_13, mô hình đề xuất cơ bản cho hiệu quả phân loại cao hơn trên các kịch bản CTU13_08 (Bảng 8), CTU13_12 (Bảng 10), CTU13_13 (Bảng 11).

Các độ đo của AE_MLP có thấp hơn nhưng không nhiều so với thuật toán MLP (Bảng 8), từ bảng ma trận nhầm lẫn (Bảng 9) cho thấy kết quả phân loại của AE_MLP là tương đương với MLP (bắt nhầm 1 bản ghi bình thường là tấn công). Mô hình AE-kNN cho hiệu suất cao nhất trong các mô hình đề xuất.

Hiệu suất của mô hình đề xuất được đánh giá dựa trên các độ đo đa số là cao hơn, tăng trong khoảng từ 0.12% đến 17.4% so với thuật toán học máy thông thường, trong khi một số độ đo (3 chỉ số) thấp hơn trong khoảng 0.42% đến 1.11% (Bảng 10, Bảng 11). Trong đó, tất cả các chỉ số của độ đo F1 đều cho hiệu suất cao hơn.

Tương tự như bộ dữ liệu UNSW-NB15, đối với tập dữ liệu CTU13_13 kiểm thử, thì sự chênh lệch giữa các bản ghi tấn công và bình thường là không nhiều, dữ liệu mất cân bằng nhẹ, nên độ đo *acc* có thể được đưa vào để so sánh đánh giá hiệu suất của các phương pháp (Bảng 11): số bản ghi normal là 6428 (44.67%), số bản ghi tấn công là 7961 (55.33%).

Do các thuật toán học máy phân loại tương đối chính xác trên tập dữ liệu này (hiệu suất từ 99.60-100%), nên mô hình đề xuất cho kết quả phân loại tương đương (thấp hơn không đáng kể khoảng 0.02%) (Bảng 12), chủ yếu là do chênh lệch ít về số lượng bắt nhầm bản ghi là tấn công. Thực nghiệm chỉ cho kết quả tốt hơn với mô hình AE_kNN (Bảng 13), hiệu suất phân loại là 100%.

Qua kết quả thực nghiệm đối với các kịch bản của Bộ dữ liệu CTU_13, có thể rút ra một kết luận là mô hình AE-kNN đều cho hiệu suất cao hơn so với các mô hình khác trên các kịch bản *CTU13_08, CTU13_12, CTU13_10.*

BẢNG 8. KẾT QUẢ PHÁT HIỆN TẤN CÔNG VỚI BỘ DỮ LIỆU CTU13_08

Phương pháp	**ML (%)**			**AE_ML (%)**		
	π	Recall	F1	π	Recall	F1
DT	84.03	**99.92**	91.29	**98.85**	99.67	**99.26**
RF	99.59	99.84	99.71	**100**	99.67	**99.84**
kNN	99.92	99.67	99.79	**99.92**	**99.94**	**99.88**
MLP	**99.92**	**99.75**	**99.83**	99.83	**99.75**	99.79
SVM	99.75	93.35	99.04	**99.92**	**99.59**	**99.75**

BẢNG 9. MA TRẬN NHẦM LẪN - CTU13_08

		Kết quả dự đoán của MLP		Kết quả dự đoán của AE_MLP	
		Normal	**Attack**	**Normal**	**Attack**
Dữ liệu thực	**Normal**	14579	**1**	14579	**2**
	Attack	**3**	1207	**3**	1207

BẢNG 10. KẾT QUẢ PHÁT HIỆN TẤN CÔNG VỚI BỘ DỮ LIỆU CTU13_12

Phương pháp	**ML (%)**			**AE_ML (%)**		
	π	Recall	F1	π	Recall	F1
DT	93.68	75.64	83.70	**98.04**	**93.04**	**95.48**
RF	98.24	**100**	99.11	**98.89**	97.66	**99.33**
kNN	**98.88**	98.88	98.88	98.45	**99.55**	**99.00**
SVM	**98.73**	90.02	94.18	98.31	**94.20**	**96.21**

BẢNG 11. KẾT QUẢ PHÁT HIỆN TẤN CÔNG VỚI BỘ DỮ LIỆU CTU13_13

Phương pháp	**ML (%)**				**AE_ML (%)**			
	acc	π	Recall	F1	acc	π	Recall	F1
DT	96.91	96.37	**98.10**	97.23	**97.73**	**98.39**	97.49	**97.94**
RF	89.56	84.78	98.88	91.29	**99.26**	**99.42**	**99.20**	**99.31**
MLP	97.74	99.07	96.82	97.93	**98.28**	**99.27**	**97.60**	**98.43**
SVM	96.48	**99.42**	94.18	96.73	**99.67**	96.17	**94.78**	**96.92**

BẢNG 12. KẾT QUẢ PHÁT HIỆN TẤN CÔNG VỚI BỘ DỮ LIỆU CTU13_10

Phương pháp	**ML (%)**			**AE_ML (%)**		
	π	**Recall**	**F1**	**π**	**Recall**	**F1**
DT	100	100	100	100	99.995	99.98
RF	100	100	100	99.98	100	99.99
Naive Bayes	99.80	99.60	99.70	99.80	99.60	99.70
kNN	100	99.995	99.998	**100**	**100**	**100**
MLP	100	99.78	99.89	100	99.78	99.89
SVM	100	99.73	99.87	100	99.72	99.86

BẢNG 13. MA TRẬN NHẦM LẪN - CTU13_10

		Kết quả dự đoán của kNN		Kết quả dự đoán của AE_kNN	
		Normal	**Attack**	**Normal**	**Attack**
Dữ liệu thực	**Normal**	3152	**0**	3152	**0**
	Attack	**1**	21287	**0**	21288

V. KẾT LUẬN

Bài báo đã trình bày kỹ thuật kết hợp mô hình học sâu AE và các bộ phân loại để phát hiện các cuộc tấn công trinh sát mạng. Thử nghiệm với bộ dữ liệu NSL-KDD, bộ dữ liệu UNSW-NB15, CTU_13 cho thấy mô hình đề xuất có tỷ lệ phát hiện và phân loại dữ liệu trinh sát mạng cao, đồng thời đảm bảo tỷ lệ báo động giả thấp hơn so với các thuật toán máy học trước đó. Đa số các mô hình đề xuất cho kết quả phân loại dữ liệu cao hơn, cho thấy việc học đặc trưng dữ liệu của AE là hiệu quả. Đối với các tập dữ liệu nhỏ, tập dữ liệu đầu vào có số chiều thấp hơn thì kết quả hiệu suất phân loại là tương đương. Dựa trên kết quả nghiên cứu thu được, hướng nghiên cứu tiếp theo sẽ thử nghiệm với các mạng dữ liệu trực tuyến để kiểm tra khả năng hoạt động của mô hình đề xuất trong các hệ thống phát hiện trinh sát tấn công mạng theo thời gian thực.

Để có được số liệu giúp hoàn thiện được bài viết này, nhóm tác giả xin cảm ơn UBND Tỉnh Vĩnh Phúc đã hỗ trợ thông qua Nhiệm vụ nghiên cứu khoa học và công nghệ cấp Tỉnh Vĩnh Phúc mã số 20/ĐTKHVP/2021-2022.

TÀI LIỆU THAM KHẢO

[1]. Elias Bou-Harb, Mourad Debbabi, and Chadi Assi, Cyber Scanning: A Comprehensive Survey, IEEE Communications Surveys and Tutorials, 2014, 16.3: 1496-1519.

[2]. Lee, S. Y, Kim, Y. S., Lee, B. H., Kang, S. H.,Youn, C. H., A probe detection model using the analysis of the fuzzy cognitive maps, Computational Science and Its ApplicationsICCSA 2005, 287-291

[3]. Vidhya. M, Efficient classification of portscan attacks using Support Vector Machine, Green High Performance Computing (ICGHPC), 2013 IEEE International Conference, 2013.

[4]. Meijuan Gao, Jingwen Tian, Mingping Xia, Intrusion Detection Method Based on Classify Support Vector Machine, secondInternational Conference on Intelligent Computation Technology and Automation, ICICTA ’09, vol. 2, pp. 391-394.

[5]. BHUYAN, Monowar H.; BHATTACHARYYA, Dhruba K.; KALITA, Jugal K, AOCD: An Adaptive Outlier Based oordinated Scan Detection Approach, IJ Network Security, 2012, 14.6: 339-351.

[6]. Nguyen Viet Hung, Nguyen Van Quan, Le Thi Trang Linh, Shone Nathan, (2018), “Using Deep Learning Model for Network Scanning Detection”, ICFET '18: Proceedings of the 4th International Conference on Frontiers of Educational Technologies , [117–121].

[7]. Van Quan Nguyen; Viet Hung Nguyen; Van Loi Cao; Nhien - An Le Khac; Nathan Shone, (2020), “Clustering-Based Deep Autoencoders for Network Anomaly Detection”, Future Data and Security Engineering (pp.290-303).

[8]. Nour Moustafa, Jill Slay, (2015), “NSW-NB15 A Comprehensive Data set for Network Intrusion Detection Systems”, School of Engineering and Information Technology, University of New South Wales at the Australian Defence Force Academy Canberra, Australia.

[9]. Xavier Glorot, Yoshua Bengio, (2010), “Understanding the difficulty of training deep feedforward neural networks”, Journal of Machine Learning Research 9, [249-256].

[10]. Thang, N. M., & Luong, T. T. (2022). Algorithm for detecting attacks on Web applications based on machine learning methods and attributes queries. Journal of Science and Technology on Information Security, 2(14), 26-34.

SƠ LƯỢC VỀ TÁC GIẢ

Nguyễn Thị Dung

Đơn vị công tác: Học viện Kỹ thuật Quân sự.
Email: ntdmta@gmail.com
Quá trình đào tạo: Học viên cao học Khóa 32 - Học viện Kỹ thuật Quân sự.
Hướng nghiên cứu hiện nay: Phát hiện tấn công mạng; trí tuệ nhân tạo.

Nguyễn Văn Quân

Đơn vị công tác: Học viện Kỹ thuật Quân sự.

Email: nguyenvanquan87@mail.ru

Quá trình đào tạo: Nhận bằng Kỹ sư tại Đại học Kỹ thuật tổng hợp Quốc gia Bauman – Matxcova năm 2012.

Hướng nghiên cứu hiện nay: Phát hiện tấn công mạng; giám sát mạng; trí tuệ nhân tạo.

Nguyễn Việt Hùng

Đơn vị công tác: Học viện Kỹ thuật Quân sự.

Email: hungnv@lqdtu.edu.vn

Quá trình đào tạo: Nhận bằng Đại học năm 2006; Thạc sĩ năm 2008 và Tiến sĩ năm 2012 tại Đại học Vật lý Kỹ thuật Matxcova.

Hướng nghiên cứu hiện nay: An toàn thông tin; phát hiện mã độc; giám sát mạng; trí tuệ nhân tạo.