ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG

Môn học : **Hệ thống Tìm kiếm, Phát hiện và Ngăn ngừa Xâm nhập**

BÁO CÁO CUỐI KỲ

Đề tài: Hybrid Intrusion Detection System for Internet of Things (IoT)

Lớp : **NT204.N21.ATCL**

Giảng viên hướng dẫn : **Đỗ Hoàng Hiển**

Nhóm 8

Nguyễn Mạnh Cường	20520421
Hoàng Văn Anh Đức	20520890
Trần Quốc Đạt	20521179
Trần Đức Minh	20521617

Thành phố Hồ Chí Minh
Ngày 28 tháng 5 năm 2023

MỤC LỤC

I. TỔNG QUAN

- Bài báo *"Hybrid Intrusion Detection System for Internet of Things (IoT)"* **[33]**, tác giả thực hiện phương pháp phát triển một hệ thống phát hiện xâm nhập (IDS) cho hệ thống IoT dựa trên mạng nơ-ron tích chập (CNN) kết hợp với mạng nơ-ron dài ngắn hạn (LSTM) *(Hybrid convolutional neural network with long short term memory)*. Phương pháp này giúp trích xuất đặc trưng và phân loại dữ liệu, nhằm giúp xác định các trạng thái tấn công và bình thường.
- Tác giả cũng sử dụng bộ dữ liệu *UNSW-NB15* để thực hiện thí nghiệm và so sánh hiệu quả của phương pháp đề xuất với phương pháp phát hiện xâm nhập dựa trên mạng nơ-ron hồi quy (Recurrent neural network). Kết quả cho thấy phương pháp đề xuất đạt hiệu quả phát hiện tốt hơn so với phương pháp truyền thống.

II. BÁO CÁO CHI TIẾT

1. Khái quát bài toán

a. Vấn đề

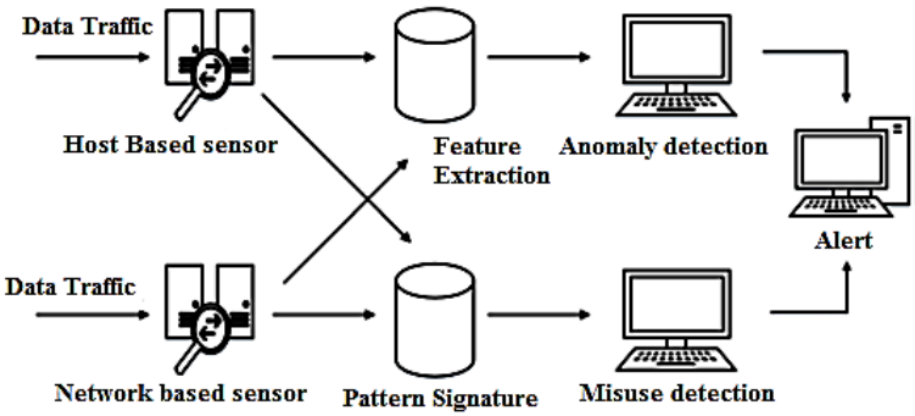
- *Internet of Things* (IoT) **[1]**, một công nghệ kết nối các thiết bị khác nhau thông qua internet, là một ứng dụng mới và hỗ trợ cho nhiều lĩnh vực như quy trình công nghiệp, chăm sóc sức khỏe, tự động hóa, môi trường thông minh,...
- Mặc dù IoT cung cấp một loạt các dịch vụ và ứng dụng, nhưng nó cũng đối mặt với nhiều vấn đề bảo mật và dễ bị tấn công do tính đa dạng và không tương thích với các phương pháp bảo mật thông thường.
- Các tính chất bảo mật chính của IoT được tóm tắt như sau:
  - *Data Confidentiality:* Quyền riêng tư dữ liệu là việc không cho phép người dùng được ủy quyền thực hiện sửa đổi hoặc thay đổi dữ liệu trên các thiết bị IoT. Nó cần thiết cho các ứng dụng công nghiệp và cá nhân, sửa đổi dữ liệu (đặc biệt trong lĩnh vực sức khỏe, y tế) sẽ dẫn đến các vấn đề nghiêm trọng.
  - *Data Integrity:* Tính toàn vẹn dữ liệu là yếu tố cần thiết trong môi trường IoT. Vì IoT hoạt động trong môi trường đa dạng và dữ liệu được di chuyển từ những nơi xa về trung tâm. Độ tin cậy và dịch vụ đảm bảo để chuyển dữ liệu từ các hệ thống xa là cần thiết trong IoT. Việc xác định tính toàn vẹn của dữ liệu bằng cách xác minh nguồn dữ liệu và xác định các cuộc tấn công độc hại trong các thiết bị, một bộ phận của hệ thống Iot.
  - *Data Availability:* Để hoạt động hiệu quả, việc truy cập dữ liệu trong môi trường IoT là rất quan trọng và cần được đảm bảo liên tục. Tuy nhiên, các thiết bị được sử dụng bởi người dùng có thể gặp phải các lỗ hổng bảo mật dẫn đến những vấn đề liên quan đến an ninh mạng.
  - *Authentication:* Nhìn chung, quá trình xác minh trong IoT khác nhau từ hệ thống này sang hệ thống khác **[2]** và việc phân loại đối tượng cần được thực hiện ở giai đoạn ban đầu **[3]** là một quá trình cần thiết. IoT yêu cầu một quá trình xác minh tốt để cung cấp cho phép truy cập và truy xuất dữ liệu kèm theo khả năng thích ứng tốt hơn. Cho nên ta phải cân bằng giữa khả năng thích ứng và bảo mật dữ liệu trong quá trình thiết kế và triển khai một môi trường IoT **[4]**.

- *Authorization:* Việc xác định quyền của người dùng đối với việc truy cập dữ liệu trong môi trường IoT cần được định nghĩa bằng quy trình ủy quyền. Bằng cách đo lường các thiết bị và tính bảo mật thông tin, quy trình ủy quyền cần được định nghĩa trong một mạng **[5]**.

- Về nhu cầu phát triển cơ chế bảo mật cho môi trường Internet of Things, tác giả nhấn mạnh rằng Data oriented security mechanism **[6]** là cần thiết để ngăn chặn truy cập trái phép đến nguồn dữ liệu từ các người dùng độc hại. Lưu ý rằng các cơ chế bảo mật thông thường dựa trên kỹ thuật mật mã không được sử dụng rộng rãi trong môi trường IoT do lượng dữ liệu lớn.

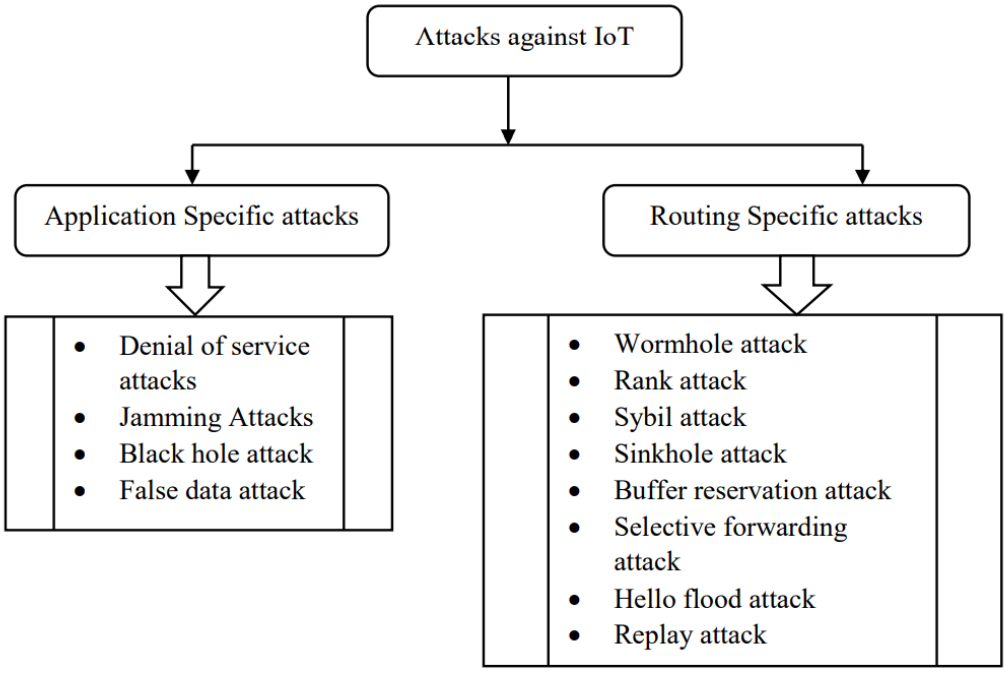
b. Hướng giải quyết

- Để đảm bảo an ninh cho mạng IoT, *hệ thống phát hiện xâm nhập* (IDS) được đề xuất nhằm nâng cao tính bảo mật và giảm thiểu các lỗ hổng bảo mật. Tác giả cho rằng xác định các mối đe dọa trong thời gian tối thiểu là quan trọng để giảm thiểu các vấn đề trong mạng. Do đó, IDS là cần thiết để xác định kẻ xâm nhập trong mạng IoT và ngăn chặn truy cập trái phép vào dữ liệu.
- Tuy nhiên, do giới hạn về tài nguyên, việc triển khai IDS có thể phức tạp. Để đơn giản hóa quá trình, một hệ thống IDS trung tâm có thể được sử dụng để giám sát mạng và các nút từ xa và kích hoạt cảnh báo cho quản trị mạng trong trường hợp có sự cố bảo mật ***(Hình 1)***.



***Hình 1**: Hệ thống phát hiện xâm nhập*

- Các hệ thống phát hiện xâm nhập được chia thành ba giai đoạn:
  - *Giám sát*, dựa trên cảm biến mạng hoặc máy chủ.
  - *Phân tích*, thực hiện trích xuất thuộc tính và quá trình nhận dạng mẫu dựa trên đó.
  - *Phát hiện*, phát hiện bất thường hoặc xâm nhập trong một mạng.
- Kiến trúc hệ thống phát hiện xâm nhập truyền thống chủ yếu tập trung vào cung cấp bảo mật cho các đặc tính quản lý internet và thiếu bảo mật dòng dữ liệu lớn thời gian thực **[7]**.
- Trong đề tài IDS, chú trọng chính vào việc phát hiện các cuộc tấn công và điều quan trọng là xác định các loại tấn công khác nhau trong môi trường IoT ***(Hình 2)***.



***Hình 2**: Các cuộc tấn công nhắm vào hệ thống IoT*

2. Phương pháp đề xuất

a. Tổng quan

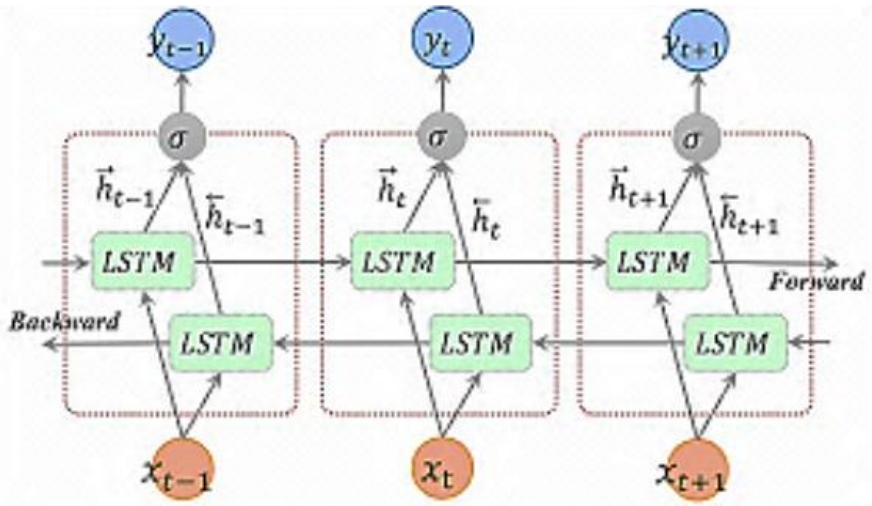
- Bài báo đề xuất một hệ thống phát hiện xâm nhập (IDS) mới cho mạng IoT dựa trên:
  - *Mô hình mạng nơ-ron tích chập* (Convolutional Neural Network - CNN);
  - *Mô hình mạng nơ-ron tái phát sinh dài ngắn hạn* (Long Short-term Memory - LSTM).
- Đề xuất nhằm giải quyết các vấn đề của các IDS hiện có, chúng thường chỉ tập trung vào phát hiện duy nhất một loại tấn công, hoặc đối với phát hiện nhiều loại tấn công thì tính toán rất tốn kém.
- Mô hình CNN **[8]** là một phương pháp deep learning phổ biến trong nhận dạng ảnh, đặc biệt là phân loại dữ liệu, trong khi mô hình LSTM **[9]** là một mô hình mạng nơ-ron tái phát sinh dùng để phân tích dữ liệu chuỗi và có khả năng giữ trạng thái trước đó để áp dụng cho dữ liệu hiện tại, từ đó giúp cải thiện hiệu suất phát hiện của hệ thống.
- Mô hình được đề xuất bao gồm bốn giai đoạn như sau:
  - *Thu thập dữ liệu:* quan trọng để tạo dữ liệu đầu vào cho mô hình. Dữ liệu là system logs và features của nó.
  - *Tiền xử lý dữ liệu:* để loại bỏ các thông tin nhiễu không cần thiết.
  - *Huấn luyện mô hình:* dữ liệu được đưa vào mô hình huấn luyện với các thông số được định nghĩa như lớp tích chập, kích thước của cửa sổ trượt, trọng số liên kết neuron và đầu ra.
  - *Xác định tấn công:* dữ liệu đã được huấn luyện và dữ liệu thực tế được xử lý cùng nhau để tính toán các trọng số và xác định các cuộc tấn công.

b. Chi tiết

- Mô hình mạng nơ-ron tích chập (CNN) đề xuất bao gồm một lớp đầu vào có tập ma trận kích thước $\mathbf{m_0}$ **x** $\mathbf{n_0}$ và lớp đầu ra có tập neuron cho mỗi nhãn **[10]**.
- *Các lớp ẩn* **[11]** được sử dụng để tổng quát hóa (trích xuất thuộc tính) các tính năng bao gồm nhiều *convolutional matrixes* **[12]** và *filter matrices* **[13]**.
- Một tập các tham số $\boldsymbol{(s, wn * hn, st1)}$ and $\boldsymbol{(s, wm * hm, st2)}$ được sử dụng để tính tích của *convolutional* **[14]** và *filter*, bao gồm *độ sâu trọng số ma trận chia sẻ* **(s) [15]** và *các bước trượt* (sliding steps) được biểu diễn là **st1, st2**.
- *Kích thước cửa sổ trượt* (Sliding window size) **[16]** là **wn x hn**.
- *Kích thước cửa sổ bộ lọc* (Filter size) **[17]** được định nghĩa là **wm x hm.**
- Lớp ẩn là lớp cuối cùng được sử dụng để kết nối lớp đầu ra, thu được kết quả phân loại.
- Các kích thước ma trận kết quả được tính dựa trên kích thước cửa sổ trượt và tích chập của matrix size, tương tự với kích thước cửa sổ bộ lọc.

$$w_n * h_n = \left(\frac{w_{n-1} - w_m}{st1_n} + 1\right) \times \left(\frac{h_{n-1} - h_m}{st1_n} + 1\right)$$

- Kích thước của resulting matrix **[19]** được đại diện bởi $\boldsymbol{w_n * h_n}$ và $\boldsymbol{w_{n-1} * h_{n-1}}$, kích thước của sliding window được biểu diễn bởi $\boldsymbol{w_m * h_m}$ và sliding step size **[18]** được cho là $\boldsymbol{st1_n}$.
- Quá trình này sử dụng LSTM để học nội dung trên toàn mạng. LSTM là một mô hình mạng nơ-ron dựa trên *Recurrent Neural Network* (RNN) được sử dụng để đưa ra đầu ra dựa trên các time stamps cho đầu vào. LSTM giúp trích xuất các thuộc tính quan trọng từ các nút và giúp xác định các nút độc hại và sự tấn công của nó.
- Tuy nhiên, việc sử dụng LSTM đơn lẻ không thể hiệu quả phát hiện xâm nhập trong mạng vì LSTM có vấn đề về gradient vanishing, không thể học thông tin trong thời gian dài. Với thời gian ngắn thì hiệu suất của LSTM tốt hơn và giúp giảm độ phức tạp của hệ thống. Do đó, LSTM được sử dụng cùng với mạng nơ-ron tích chập trong hệ thống phát hiện xâm nhập đề xuất.



***Hình 3**: Kiến trúc Bidirectional LSTM*

- Mô hình Bidirectional LSTM được mô tả trong **Hình 3**, trong đó một chuỗi đầu vào được chuyển tiếp đến các lớp ẩn của mạng và tạo ra đầu ra trong lớp đầu ra tương ứng. Sau đó, các mô hình LSTM hai chiều được phát triển để xử lý chuỗi đầu vào theo hướng thuận và ngược bằng hai lớp ẩn. Những tính năng này liên quan đến truyền dữ liệu trong mạng và thu được dữ liệu cần thiết.
- Trong giai đoạn huấn luyện mô hình:
  - Các thuộc tính đầu vào được đánh nhãn và mỗi phần tử được gán trong ma trận thuộc tính dựa trên các neuron đầu vào. Sau khi xử lý đầu vào, mạng tính toán trọng số liên kết neural và ma trận trọng số cho lớp tích chập.
  - Hệ thống có ***x*** *lớp tích chập* và *các lớp vector được biểu diễn dưới dạng* ***f***, các *neuron đầu ra được biểu diễn dưới dạng* ***y*** và hàm trọng số được thu được là:

$$W_y = \sum_{n=1}^{N} \sum_{m=1}^{M_n} (x_n * f_n + 1) + x * f$$

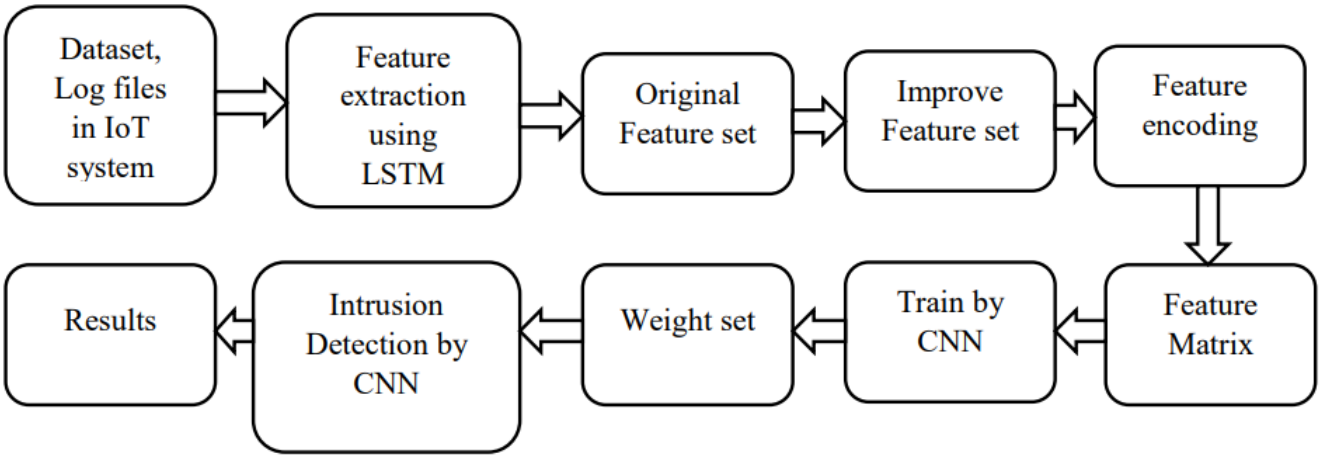
    - W_y : tổng trọng số của mạng.
    - x_n * f_n : kích thước window filter.
    - M_n : số lượng filter ở lớp tích chập.
  - Trong quá trình huấn luyện , mỗi đầu vào và trọng số **[20]** của nó tạo ra labels dựa trên *hàm mất mát* **[21]**. Hàm mất mát thu được dựa trên *compactness và descriptiveness* **[22]** của dữ liệu xác thực và đào tạo mô hình và nó được đưa ra dưới dạng:

$$L_f = L_c + \varphi * L_d$$

    - φ : hệ số tỷ lệ được sử dụng để đánh giá mức độ ưu tiên của nút
    - L_c , L_d : đại diện cho compactness loss và descriptiveness loss
- Trong giai đoạn phát hiện tấn công:
  - Các thuộc tính được trích xuất từ tập dữ liệu và sau đó được chuyển đổi thành *feature matrix* **[23]**. Nó giúp lấy tập trọng số từ quá trình huấn luyện và *output layers defines neural label* **[24]**.
  - Sử dụng *activation function* **[25]**, các kết quả đầu ra quan sát được và được đưa ra dưới dạng:

$$f_m(y) = \frac{e^{om}}{\sum_{m-1}^{n} e^{on}}$$

  - Activation function lấy đầu ra xác suất của mỗi neuron và trọng số của các *fully connected layer weigh* **[26]**.
  - Thông thường, giá trị của *activation function* nằm trong khoảng từ 0 đến 1 và giá trị tối đa đại diện cho nhãn đầu ra.



***Hình 4**: Tổng quan đề xuất*

- Tổng quan đề xuất (**Hình 4**):
  - Trong giai đoạn huấn luyện, dataset và các logs của nó được thu thập để trích xuất các thuộc tính cần thiết, và so sánh với bộ thuộc tính ban đầu để cải thiện các thuộc tính của dữ liệu hiện có bằng cách sử dụng LSTM.
  - Sau khi xác định *label function* **[27]**, dữ liệu được lựa chọn và huấn luyện bằng CNN.
  - Dựa trên *weight function* **[28]** được thu được sau quá trình huấn luyện CNN, các cuộc tấn công được phân loại thành các kết quả.

3. Thực nghiệm

a. Thực nghiệm trong bài báo

- Trong bài báo, tác giả thực nghiệm đề xuất của mình và so sánh kết quả với một mô hình IDS dựa trên mạng nơ-ron hồi quy (Recurrent neural network - RNN).
  - Tập dữ liệu *UNSW-NB15* (bao gồm *240 mẫu bình thường* và *3890 mẫu tấn công*) được sử dụng với tỷ lệ xác thực *70% cho việc huấn luyện* và *30% cho việc kiểm tra*.
  - Hệ thống đề xuất trích xuất đặc trưng từ tập dữ liệu và xác định các điều kiện tấn công và bình thường. Mô hình được thực nghiệm bằng *Tensorflow* được cài đặt trên bộ vi xử lý *Intel i5 2.4GHz* với *8GB RAM*.
- **Bảng 1** mô tả giá trị trung bình của các chỉ số đánh giá sử dụng để so sánh mô hình đề xuất và mô hình RNN. Từ giá trị này, quan sát được rằng mô hình đề xuất đạt được hiệu suất phát hiện *tốt hơn* so với mô hình RNN. Một số tham số như recall và precision *tương tự* như RNN, tuy nhiên tỷ lệ true positive và false positive của mô hình đề xuất *tốt hơn rất nhiều*. Accuracy đạt được 98%, cao hơn 3% so với mô hình RNN truyền thống.

S.No	Parameter	RNN	Proposed HCNN
1	Precision	1	1
2	Recall	0.99	1
3	f-score	0.98	0.99
4	Miscalculation rate	0.041	0.032
5	Detection time (sec)	2.19	1.88
6	Accuracy (%)	95.7	98.6

Bảng 1: *So sánh đề xuất với RNN-based IDS*

b. Thực nghiệm nhóm triển khai

- Bởi vì tác giả không công bố mã nguồn mở cho đề xuất của mình, nhóm đã cố gắng triển khai thực nghiệm giống với phương pháp nhất có thể. Nhóm cũng sử dụng tập dữ liệu *UNSW-NB15*, các thông số cho mô hình dựa theo triển khai của bài báo.
- *Chi tiết từng bước triển khai và giải thích vui lòng theo dõi video demo ở liên kết đính kèm.*

4. So sánh bài báo

- Tiêu chí nhóm lựa chọn 2 bài báo sau để so sánh là các phương pháp đề xuất ứng dụng học máy vào IDS cho IoT và có chung tập dữ liệu *UNSW-NB15* với bài báo chính. Tuy nhiên, với việc lựa chọn thông số, môi trường triển khai khác nhau, cùng việc tiền xử lý cho tập dữ liệu khác nhau, *các thông số cho kết quả thực nghiệm chỉ mang tính chất tham khảo*. So sánh tập trung vào việc cách bài báo ứng dụng các phương pháp vào bài toán vấn đề.

a. Bài báo 1

- Bài báo 1, *"Deep-Intrusion Detection System with Enhanced UNSW-NB15 dataset Based on Deep Learning Techniques"* **[31]** đề xuất một mô hình IDS cho mạng IoT dựa trên Deep Learning.
  - Bài báo thực hiện một số tối ưu cho tập dữ liệu *UNSW-NB15*, khiến tập có thể dễ dàng thử nghiệm với 3 mô hình tác giả đề xuất sau đây. Ngoài ra, còn cải thiện tập dữ liệu để có thể áp dụng cho các mô hình *phân loại đa lớp* (multi-classification) bằng cách kết hợp 4 tệp *.CSV* lại thành một tệp.
  - Cả 3 mô hình riêng lẻ mà tác giả đề xuất đều thuộc phương pháp Deep Learning, cùng với CNN và LSTM của bài báo chính. Cụ thể là: *Artificial Neural Network* (ANN);

Deep Neural Network (DNN); và *Recurrent Neural Network* (RNN) *with Long Short-Term Memory* (LSTM).

- Thực nghiệm triển khai với tập dữ liệu và cả 3 mô hình trên, trong 2 trường hợp *binary classification* và *multi-classification*. Tỉ lệ huấn luyện mô hình là *70% training, 15% validation và 15% testing*.

- Kết quả công bố được tổng quát ở **Bảng 2**. Tuy nhiên, như đã nói ở trên, với phần tiền xử lý và một số thông số thực nghiệm khác với bài báo chính, kết quả đánh giá chỉ mang mục đích tham khảo.

Machine Learning type	Acc. In Binary	Acc. In Multi-class
Proposed ANN	99.26 %	97.89 %
Proposed DNN	99.22 %	99.59 %
Proposed RNN-LSTM	85.42 %	85.38 %

***Bảng 2**: Kết quả thực nghiệm của bài báo so sánh 1*

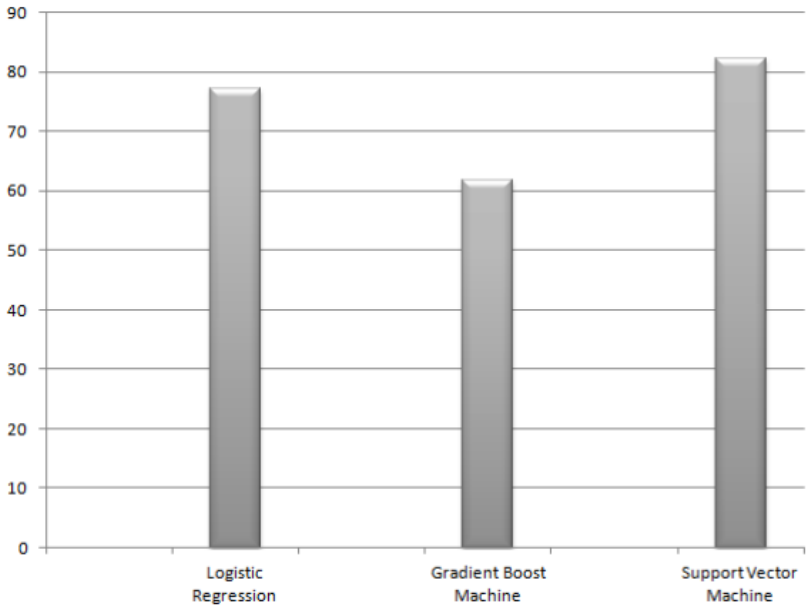
- Ưu điểm của bài báo 1 so với bài báo chính là:
  - Có tối ưu được tập dataset để phụ hợp hơn với các mô hình Deep Learning, cũng như xử lý nhãn cho phù hợp với các mô hình *phân loại đa lớp* (có thể phân biệt các loại tấn công khác nhau và bình thường, so với bài báo chính chỉ có thể phần loại nhị phân – bình thường/tấn công).
  - Triển khai cả *binary classification* và *multi-classification*; có thêm bước *validation* cho quá trình huấn luyện mô hình, giúp cải thiện hiệu xuất và tránh sự trùng lập, sai xót ở tập dữ liệu đầu vào.
- Song, bài báo chính cũng có ưu điểm khi áp dụng được một đề xuất cho việc kết hợp các phương pháp học máy (CNN-LSTM) trong mô hình IDS cho IoT.

b. Bài báo 2

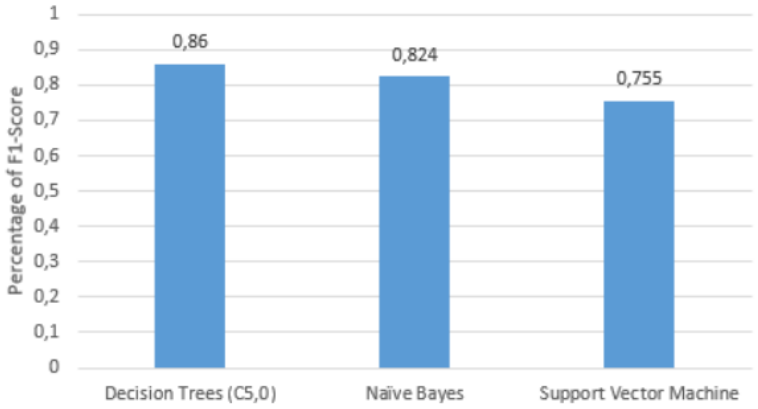
- Bài báo 2, *"Network Based Intrusion Detection Using the UNSW-NB15 Dataset"* **[32]** đề xuất một mô hình anomaly-based network IDS cho IoT hai giai đoạn dựa trên học máy.
  - Với tập *UNSW-NB15*, tác giả sử dụng Recursive Feature Elimination và Random Forests cùng các kỹ thuật khác để chọn ra các đặc trưng tốt nhất của tập dữ liệu cho mục đích học máy.
  - Sau đó, thực hiện *phân loại nhị phân* (binary classification) để xác định lưu lượng xâm nhập so với lưu lượng bình thường. Các mô hình phân loại riêng lẻ bao gồm *Logistic Regression, Gradient Boost Machine (GBM) và Support Vector Machine (SVM)* là những mô hình học máy truyền thống, không thuộc phương pháp sử dụng mạng nơ-ron nhân tạo (ANN) hay các kỹ thuật Deep Learning như bài báo chính.
  - Đầu ra của SVM (đạt kết quả tốt nhất ở bước trước) được đưa vào một loạt bộ *phân loại đa lớp* để cải thiện độ chính xác trong việc dự đoán loại tấn công. Cụ thể là *Decision Trees (C5.0), Naïve Bayes và multinomial Support Vector Machine*.

- Triển khai thực nghiệm với tỉ lệ *70% training, 30% testing*. Kết quả cho thấy khi kết hợp hai giai đoạn, kết quả hiệu suất của mô hình gia tăng 12%.

- **Hình 5** cho thấy kết quả triển khai mô hình được tác giả công bố. Tuy như đã nói ở trên về kết quả thực nghiệm chỉ mang tính chất tham khảo, nhưng có thể thấy nhìn chung các số liệu đánh giá có phần thấp hơn bài báo 1 và bài báo chính.



(a)



(b)

Hình 5: *Kết quả thực nghiệm của bài báo so sánh 2*
(a) *Giai đoạn 1 (Accuracy)* ***(b)*** *Giai đoạn 2 (F1-Score)*

- Ưu điểm của bài báo 2 so với bài báo chính là:
  - Cũng như bài báo 1, tác giả có tối ưu đề xuất của mình để phân loại đa lớp các tấn công và bình thường.
  - Đề xuất 2 giai đoạn giúp cho việc chọn lựa được mô hình nào mang lại kết quả tốt nhất, cũng như tối ưu phân loại đa lớp.
- Tuy nhiên, với một môi trường có quy mô và lượng dữ liệu lớn như IoT, các mô hình ứng dụng các kỹ thuật Deep Learning (bài báo chính) sẽ mang lại hiệu quả tốt hơn.

5. Đề xuất cải thiện

- *Mở rộng kiến trúc CNN-LSTM*: có thể xây dựng một mô hình phức tạp hơn bằng cách thêm các lớp CNN và LSTM bổ sung. Bằng cách tăng độ sâu và độ rộng của mạng, mô hình có thể học được các mẫu phức tạp và trừu tượng hơn từ dữ liệu IoT.
- *Sử dụng ảnh đầu vào*: Thay vì sử dụng các đặc trưng truyền thống như dữ liệu dạng chuỗi thời gian, có thể chuyển đổi dữ liệu IoT thành các hình ảnh để sử dụng CNN. Các kỹ thuật như SPECTRO-IoT đã được đề xuất để biến đổi dữ liệu IoT thành hình ảnh để sử dụng trong mô hình CNN-LSTM.
- *Sử dụng kỹ thuật Attention*: **[29]** Cấu trúc mạng CNN-LSTM có thể được kết hợp với kỹ thuật attention để tăng cường khả năng tập trung vào các phần quan trọng của dữ liệu IoT. Attention cho phép mô hình tập trung vào các đặc trưng quan trọng nhất trong quá trình phát hiện xâm nhập.
- *Kết hợp với các mô hình học sâu khác*: Thay vì chỉ sử dụng CNN và LSTM, có thể kết hợp mô hình IDS này với các mô hình học sâu khác như Gated Recurrent Unit (GRU), Transformer, hay các mạng học sâu khác để tăng cường khả năng phát hiện và phân loại xâm nhập trong hệ thống IoT.
- *Data augmentation*: **[30]** Có thể sử dụng các kỹ thuật tăng cường dữ liệu (data augmentation) để tăng độ đa dạng của dữ liệu đầu vào. Việc này có thể giúp cho mô hình đào tạo được hiệu quả hơn và giảm thiểu hiện tượng overfitting.

III. KẾ HOẠCH THỰC HIỆN

Thời gian	Đảm nhận	Nội dung
20/03/2023 – 02/04/2023	Tất cả các thành viên	Tự đọc sơ lược bài báo.
03/04/2023 – 09/04/2023	Anh Đức, Quốc Đạt	Tìm nguồn tài liệu, cách triển khai thực nghiệm.
	Mạnh Cường, Đức Minh	Làm báo cáo giữa kỳ.
10/04/2023 – 30/04/2023	Anh Đức, Quốc Đạt	Triển khai thực nghiệm.
	Mạnh Cường, Đức Minh	Nghiên cứu phương pháp của bài báo.
01/05/2023 – 28/05/2023	Mạnh Cường	Viết báo cáo, làm slides thuyết trình.
	Anh Đức	Viết báo cáo.
	Quốc Đạt	Làm video demo, mô tả thực nghiệm.
	Đức Minh	Thuyết trình.

IV. PHỤ LỤC VÀ TRÍCH DẪN

- **[1] IoT module** là một thiết bị hoặc một bộ phận của hệ thống IoT, được sử dụng để kết nối các thiết bị đo, cảm biến hoặc các thiết bị khác với mạng IoT và cho phép chúng truyền thông với nhau và với các trung tâm điều khiển.
  - Module IoT thường có các tính năng như kết nối mạng không dây, tích hợp các cảm biến và bộ xử lý, hỗ trợ các giao thức truyền thông và các tính năng bảo mật.
  - Các module IoT có thể được sử dụng trong nhiều ứng dụng khác nhau như trong các thiết bị đeo tay thông minh, các hệ thống tự động hóa nhà thông minh, các thiết bị y tế thông minh, và các hệ thống giám sát công nghiệp.
- **[2]** Một ví dụ về **quá trình xác minh trong IoT khác nhau từ hệ thống này sang hệ thống khác** là quá trình xác minh thông qua mã PIN (Personal Identification Number) trên một thiết bị IoT, ví dụ như một hệ thống an ninh nhà thông minh.
  - Một hệ thống nhà thông minh sẽ yêu cầu người dùng nhập mã PIN để mở khóa cửa hoặc tắt hệ thống báo động.
  - Tuy nhiên, trong một hệ thống khác như hệ thống y tế thông minh, xác minh sẽ được thực hiện bằng cách sử dụng các thông tin y tế của người dùng để đảm bảo rằng chỉ có các bác sĩ hoặc nhân viên y tế được phép truy cập và xem thông tin của bệnh nhân.
  - Do đó, quá trình xác minh trong IoT sẽ phụ thuộc vào mục đích và tính chất của hệ thống IoT được sử dụng.

- **[3]** Ví dụ về việc **phân loại đối tượng trong IoT** có thể là trong hệ thống quản lý giao thông thông minh, các đối tượng có thể được phân loại là ô tô, xe máy, xe đạp, người đi bộ, v.v.
  - Các loại đối tượng này cần được phân loại để đảm bảo rằng các thiết bị IoT có thể thu thập dữ liệu chính xác về số lượng, vị trí và tốc độ di chuyển của từng đối tượng.
  - Việc phân loại đối tượng được thực hiện ở giai đoạn ban đầu của quá trình IoT giúp cải thiện khả năng hoạt động của hệ thống và tăng tính đáng tin cậy của dữ liệu thu thập được.
- **[4] "Tradeoff between the IoT environment and data in the system"** có nghĩa là cân bằng giữa môi trường IoT và dữ liệu trong hệ thống.
  - Khi triển khai IoT, cần đảm bảo môi trường IoT được hoạt động một cách an toàn, ổn định và hiệu quả, đồng thời đảm bảo dữ liệu được bảo mật, truy cập được và sử dụng một cách hiệu quả.
  - Khả năng thích ứng (adaptability) trong IoT là khả năng của các thiết bị IoT để thích nghi và thay đổi trong môi trường động, bao gồm khả năng tự động cấu hình, tái cấu trúc và phát hiện lỗi. Điều này cho phép các thiết bị IoT hoạt động tốt trong các điều kiện thay đổi và đa dạng và tăng cường tính linh hoạt của hệ thống..
- **[5] Quy trình ủy quyền** (authorization process) trong một mạng IoT là quá trình xác định các quyền truy cập cho người dùng đối với dữ liệu trong môi trường IoT.
  - Quá trình này cần được xác định một cách chính xác dựa trên đánh giá mức độ tin cậy của các thiết bị IoT và độ bảo mật của thông tin được truyền.
  - Khi quy trình ủy quyền được định nghĩa tốt, người dùng sẽ chỉ có thể truy cập và sử dụng dữ liệu mà họ có quyền truy cập. Quy trình này đóng một vai trò quan trọng trong việc bảo vệ mạng IoT khỏi các cuộc tấn công và đảm bảo tính riêng tư và bảo mật của dữ liệu.A
- **[6] Data-oriented security mechanism** là một cơ chế bảo mật tập trung vào bảo vệ dữ liệu trong hệ thống và ngăn chặn truy cập trái phép đến nguồn dữ liệu từ các kẻ tấn công.
  - Cơ chế này tập trung vào bảo vệ tính bảo mật, toàn vẹn và sẵn sàng của dữ liệu, bao gồm các hoạt động như mã hóa, chứng thực, kiểm soát truy cập và giám sát dữ liệu.
  - Data-oriented security mechanism là một trong những phương pháp bảo mật quan trọng trong môi trường IoT, nơi lượng dữ liệu lớn và đa dạng tạo ra những thách thức riêng biệt đối với việc bảo mật.
- **[7] "Large volume data streams"** (luồng dữ liệu lớn) là thuật ngữ chỉ các lượng dữ liệu lớn và liên tục được tạo ra từ các nguồn khác nhau và được truyền tải trong thời gian thực qua mạng.

  - Ví dụ về luồng dữ liệu lớn bao gồm các trang web phương tiện xã hội như Twitter hoặc Facebook, các trò chơi trực tuyến, các cảm biến IoT và các hệ thống giám sát mạng. Để xử lý các luồng dữ liệu lớn, các hệ thống phải có khả năng xử lý và phân tích dữ liệu nhanh chóng và hiệu quả để phát hiện các sự cố hoặc mối đe dọa đến bảo mật
- **[8] Mô hình CNN** (Convolutional Neural Network) là một trong những kiểu mô hình được sử dụng nhiều trong học sâu (deep learning) để xử lý các tác vụ nhận dạng ảnh, âm thanh, văn bản và video.
  - Mô hình này sử dụng (convolutional layers) để học các đặc trưng (features) của dữ liệu đầu vào thông qua việc trích xuất các thông tin quan trọng từ các vùng nhỏ của ảnh hay video.
  - Sau đó, các lớp pooling được sử dụng để giảm kích thước của dữ liệu đầu ra, giúp giảm độ phức tạp tính toán và cải thiện khả năng tổng quát hóa của mô hình.
  - Cuối cùng, các lớp kết nối đầy đủ (fully connected layers) được sử dụng để kết hợp các đặc trưng đã được học để phân loại dữ liệu đầu vào. Mô hình CNN đã được áp dụng thành công trong nhiều lĩnh vực như xử lý ảnh, nhận dạng giọng nói và xử lý văn bản tự nhiên.
- **[9] LSTM** (Long Short-Term Memory) là một loại mô hình mạng nơ-ron dựa trên mạng RNN (Recurrent Neural Network), được thiết kế để xử lý các chuỗi dữ liệu có thể dài hoặc ngắn.
  - LSTM có khả năng giữ lại thông tin quan trọng và loại bỏ thông tin không quan trọng trong quá trình xử lý chuỗi, giúp cho việc dự đoán hoặc phân loại trên dữ liệu chuỗi trở nên hiệu quả hơn. LSTM có thể được sử dụng cho nhiều nhiệm vụ khác nhau, chẳng hạn như: xử lý ngôn ngữ tự nhiên, dự đoán chuỗi thời gian, phân tích dữ liệu chuỗi thị giác, và nhiều hơn nữa.
- **[10] "Set of neurons for each labels"** đề cập đến việc sử dụng một nhóm các neuron để đại diện cho một tập hợp các nhãn khác nhau.
  - Khi huấn luyện một mô hình mạng neuron, mỗi nhãn sẽ có một nhóm các neuron tương ứng để đưa ra dự đoán.
  - Ví dụ, trong một mô hình phân loại hình ảnh với 10 lớp khác nhau, sẽ có một tập hợp các neuron ở lớp đầu ra đại diện cho từng lớp khác nhau, tức là có 10 nhóm neuron ở lớp đầu ra. Khi một hình ảnh được đưa vào mô hình, mỗi nhóm neuron sẽ đưa ra một giá trị tương ứng, thể hiện độ chắc chắn của mô hình về xác suất hình ảnh đó thuộc về lớp nào.
- **[11] Hidden layer** (lớp ẩn) là một lớp trong kiến trúc của mạng neural, nằm giữa lớp đầu vào và lớp đầu ra.
  - Lớp ẩn nhận đầu vào từ lớp đầu vào, xử lý thông tin và tạo ra đầu ra cho lớp đầu ra. Chức năng của lớp ẩn là trích xuất các đặc trưng từ dữ liệu đầu vào, giúp mô hình học được mối quan hệ phức tạp giữa các đặc trưng và đưa ra kết quả dự đoán

chính xác hơn. Một mạng neural có thể có nhiều lớp ẩn, tùy thuộc vào độ phức tạp của mô hình và độ phức tạp của dữ liệu đầu vào.

- **[12] Convolutional matrix** (ma trận tích chập) là một phương pháp xử lý ảnh trong mạng deep learning.
  - Nó được sử dụng để giảm số lượng tham số và tính toán cần thiết để huấn luyện mạng.
  - Convolutional matrix được sử dụng để tìm kiếm các đặc trưng của ảnh bằng cách áp dụng một ma trận (hay kernel) có kích thước nhỏ lên toàn bộ ảnh. Kết quả của phép tích chập này sẽ tạo ra một ảnh mới, với các đặc trưng được tách ra rõ ràng hơn.
  - Các convolutional matrix khác nhau có thể được sử dụng để tìm kiếm các đặc trưng khác nhau của ảnh, chẳng hạn như cạnh, góc, hay các đặc trưng phức tạp hơn như vật thể.
- **[13] Filter matrices** (còn được gọi là kernel hoặc filter) trong mạng CNN (Convolutional Neural Network) là một ma trận dùng để phát hiện và trích xuất các đặc trưng (features) của dữ liệu đầu vào.
  - Quá trình tích chập (convolution) được thực hiện bằng cách trượt filter qua các vùng của dữ liệu đầu vào để tạo ra các feature maps, từ đó sử dụng các phép biến đổi tiếp theo để rút trích thông tin hữu ích.
  - Filter matrices thường được khởi tạo ngẫu nhiên ban đầu và được cập nhật trong quá trình huấn luyện mạng để tối ưu hóa quá trình trích xuất đặc trưng.
- **[14] Tích chập** (Convolution) là một phép toán trong xử lý tín hiệu và xử lý ảnh, được sử dụng để trích xuất thông tin từ các tín hiệu đầu vào. Trong lĩnh vực trí tuệ nhân tạo, tích chập là một phép toán quan trọng trong mạng neuron nhân tạo sử dụng để phân tích hình ảnh, âm thanh, văn bản và các loại dữ liệu khác.
  - Trong xử lý ảnh, phép tích chập thực hiện việc "trượt" một mặt nạ (kernel) trên toàn bộ bức ảnh đầu vào, tính toán tích chập giữa kernel và từng vùng của ảnh để tạo ra một bức ảnh mới với các đặc trưng được trích xuất từ ảnh gốc. Các đặc trưng này có thể là cạnh, góc, đường cong, điểm nổi bật, và các đặc trưng khác phù hợp với mục đích của bài toán.
  - Kernel được trượt (đi qua) trên bức ảnh từng pixel một, tính toán giá trị tích chập tại mỗi vị trí trên ảnh đầu vào. Khi kernel được trượt trên ảnh, mỗi vị trí trên ảnh sẽ được gán giá trị mới bằng tích chập của kernel với phần ảnh tương ứng.
  - Kết quả của phép tích chập sẽ tạo ra một bức ảnh mới, nơi mà mỗi điểm ảnh trên bức ảnh mới tương ứng với kết quả tích chập của kernel với vùng ảnh tương ứng trên bức ảnh gốc.
  - Quá trình trượt kernel trên bức ảnh này cho phép trích xuất các đặc trưng của ảnh, như cạnh, góc, đường cong, điểm nổi bật và các đặc trưng khác. Tùy thuộc vào kích thước và giá trị của kernel, kết quả của phép tích chập có thể giúp chúng ta

giảm nhiễu, tăng cường độ tương phản và giúp cho quá trình nhận dạng ảnh trở nên hiệu quả hơn.

- **[15] Shared weight matrices depth** là một khái niệm liên quan đến kiến trúc của mạng nơ-ron tích chập (CNN). Trong các mạng CNN, mỗi lớp tích chập sử dụng một bộ trọng số để thực hiện việc tích chập trên ảnh đầu vào và tạo ra các đặc trưng mới.
  - Khi số lượng lớp tích chập trong mạng tăng lên, số lượng tham số cần tối ưu cũng tăng theo, dẫn đến khó khăn trong việc huấn luyện và khả năng bị overfitting.
  - Để giảm số lượng tham số cần tối ưu trong mạng CNN, ta có thể sử dụng kỹ thuật shared weight matrices depth.
  - Theo cách này, các lớp tích chập trong mạng chia sẻ cùng một bộ trọng số để thực hiện việc tích chập trên ảnh đầu vào.
  - Khi áp dụng kỹ thuật này, mỗi lớp tích chập chỉ sử dụng một phần của bộ trọng số được chia sẻ và phần còn lại được sử dụng bởi các lớp tích chập khác có cùng shared weight matrices depth.
  - Ví dụ, nếu shared weight matrices depth là 2, các lớp tích chập trong mạng CNN sẽ chia sẻ cùng một bộ trọng số ở 2 lớp liên tiếp của mạng.
  - Với cách thiết kế này, số lượng tham số cần tối ưu trong mạng sẽ giảm đi đáng kể, giúp tăng tốc độ tính toán và giảm khả năng overfitting. Tuy nhiên, việc thiết kế shared weight matrices depth cần được cân nhắc kỹ lưỡng để đảm bảo độ chính xác và hiệu suất của mạng.
  - Ví dụ, nếu chúng ta có một lớp tích chập với 32 bộ lọc và kích thước mỗi bộ lọc là 3x3, thì số lượng tham số sẽ là 32 x 3 x 3 = 288. Nếu chúng ta có thêm một lớp tích chập khác với số lượng bộ lọc và kích thước tương tự, nhưng ta sử dụng shared weight matrices depth để chia sẻ cùng một bộ trọng số giữa 2 lớp tích chập này, thì số lượng tham số cần tối ưu chỉ là 32 x 3 x 3 = 288, thay vì 2 x 288 = 576.
- **[16] Kích thước cửa sổ trượt** là kích thước của cửa sổ được trượt qua toàn bộ ảnh đầu vào
  - Được thực hiện bằng cách chọn một khu vực trên ảnh đầu vào có kích thước cố định, rồi sau đó trượt trên đó với một bước nhảy nhất định để tạo ra các khung hình con (sub-image) của ảnh
  - Kích thước cửa sổ trượt có ảnh hưởng trực tiếp đến độ phân giải và khả năng phát hiện các đối tượng trong ảnh.
  - Khi kích thước filter càng lớn, độ phân giải của các khung hình con tạo ra sẽ giảm và khả năng phát hiện các đối tượng có kích thước nhỏ sẽ giảm đi.
  - Tuy nhiên, khi kích thước cửa sổ trượt càng nhỏ, số lượng khung hình con tạo ra cũng càng nhiều, dẫn đến tăng độ phức tạp tính toán.
  - Do đó, kích thước cửa sổ trượt thường được lựa chọn sao cho phù hợp với kích thước và số lượng đối tượng cần phát hiện trong ảnh.

- **[17] Kích thước cửa sổ bộ lọc** được sử dụng trong các thuật toán xử lý ảnh là các ma trận nhỏ được xác định trước kích thước.
    - Kích thước của các bộ lọc cần sử dụng trong một bài toán cụ thể thường được chọn và điều chỉnh dựa trên mục đích và độ phức tạp của bài toán.
- **[18] Sliding step size** (kích thước bước trượt) là kích thước khoảng cách giữa các vị trí khung hình con (sub-image) trên ảnh khi thực hiện sliding window (cửa sổ trượt) trong các mô hình xử lý ảnh.
    - Khi thực hiện sliding window, cửa sổ trượt sẽ được di chuyển qua từng vị trí trên ảnh với một khoảng cách bằng sliding step size, tạo ra các khung hình con (sub-image) tương ứng để xử lý.
    - Giá trị sliding step size sẽ ảnh hưởng đến kích thước của các khung hình con được tạo ra và ảnh hưởng đến hiệu quả và tốc độ xử lý của mô hình.
- **[19] Resulting matrix size** là kích thước ma trận kết quả thu được sau khi thực hiện phép tích chập (convolution) giữa ma trận đầu vào (input matrix) và ma trận trọng số (weight matrix).
    - Ma trận trọng số (weight matrix) là một ma trận số được sử dụng trong các mô hình máy học để biểu diễn mối quan hệ giữa các đặc trưng của dữ liệu đầu vào và đầu ra tương ứng.
    - Trong các mô hình neural network, ma trận trọng số được sử dụng để tính toán trọng số và lỗi, cũng như để thực hiện các phép toán trên dữ liệu đầu vào để tạo ra dữ liệu đầu ra. Trong quá trình huấn luyện mô hình, các giá trị trong ma trận trọng số được cập nhật dựa trên độ lỗi và các phương pháp tối ưu hóa để tìm ra các giá trị tối ưu nhất cho mô hình.
- **[20] Trọng số của đầu vào** (input weights) là các tham số trong mạng nơ-ron sử dụng để biến đổi các đặc trưng đầu vào thành các đặc trưng ở các lớp tiếp theo. Trong quá trình huấn luyện, các trọng số của đầu vào sẽ được điều chỉnh để tối ưu hoá kết quả dự đoán của mô hình.
- **[21] Hàm mất mát** (loss function) là một hàm số đo lường sự khác biệt giữa kết quả dự đoán của mô hình với giá trị đích (ground truth) mong đợi. Hàm mất mát thường được sử dụng để tối ưu các tham số trong mô hình.
    - Trong quá trình huấn luyện, mô hình sẽ điều chỉnh các tham số của mình để giảm thiểu giá trị của hàm mất mát, từ đó cải thiện khả năng dự đoán của mô hình trên tập dữ liệu huấn luyện và trên các tập dữ liệu mới.
    - Các loại hàm mất mát phổ biến trong machine learning bao gồm: mean squared error (MSE), mean absolute error (MAE), cross-entropy loss, và hinge loss. Các loại hàm mất mát này được sử dụng tùy theo loại bài toán và mục đích của mô hình.
- **[22] Compactness** (tính gọn) đo lường độ phân tán của các điểm dữ liệu trong một nhóm. Các đặc trưng có tính gọn cao sẽ có các điểm dữ liệu gần nhau hơn, tạo ra một

phân bố đồng đều và giảm thiểu số lượng các đặc trưng lặp lại. Compactness giúp giảm thiểu sự phụ thuộc của các đặc trưng vào các đặc trưng khác, từ đó giảm thiểu sự quá khớp và tăng khả năng tổng quát hóa của mô hình.
    - **Descriptiveness** (tính mô tả) đo lường khả năng của các đặc trưng để phân biệt các điểm dữ liệu khác nhau. Các đặc trưng có tính mô tả cao sẽ có giá trị khác biệt lớn giữa các điểm dữ liệu khác nhau và tạo ra một phân bố phân tán. Tính mô tả giúp tăng độ chính xác và độ tin cậy của mô hình.
    - Tổng quát lại, các đặc trưng tốt là các đặc trưng có tính gọn cao và tính mô tả cao, giúp trích xuất được các thông tin quan trọng từ dữ liệu và tăng tính tổng quát hóa của mô hình.
- **[23] Feature matrix** là một ma trận được tạo ra bằng cách sắp xếp các đặc trưng của các mẫu dữ liệu trong các hàng và các thuộc tính của các đặc trưng trong các cột.
    - Feature matrix là đầu vào cho các thuật toán học máy và được sử dụng để huấn luyện và kiểm tra các mô hình học máy
- **[24]** Trong mạng neuron, **output layer** là lớp cuối cùng của mạng được sử dụng để tính toán đầu ra của mô hình cho một đầu vào cụ thể.
    - Khi các đầu vào được đưa vào mạng, chúng được xử lý thông qua các lớp trung gian để trích xuất đặc trưng và cuối cùng đến output layer để tính toán đầu ra.
    - Ví dụ, trong bài toán phân loại hình ảnh, output layer có thể được thiết kế với số lượng neuron bằng với số lượng lớp hình ảnh cần phân loại.
    - Mỗi neuron trong output layer sẽ đại diện cho một lớp cụ thể, và giá trị đầu ra của neuron đó sẽ cho biết xác suất của hình ảnh đó thuộc về lớp tương ứng.
- **[25] Activation function** (hàm kích hoạt) là một hàm được áp dụng trên output của một neuron để đưa ra output của layer đó.
    - Cụ thể, nó tính toán tổng trọng số đầu vào của neuron và áp dụng một phép tính toán phi tuyến để đưa ra output của neuron đó.
    - Điều này cho phép mô hình học các quan hệ phi tuyến trong dữ liệu và là một phần quan trọng trong thiết kế của các mạng neural. Các hàm kích hoạt phổ biến bao gồm Sigmoid, Tanh, ReLU, Leaky ReLU, Softmax, vv.
- **[26] Fully connected layer** (hay còn gọi là dense layer) là một loại layer trong mạng neuron nhân tạo, trong đó mỗi neuron của layer này được kết nối với tất cả các neuron của layer trước đó (nếu có).
    - Nó được gọi là "fully connected" vì các neuron ở layer này có kết nối với tất cả các neuron ở layer trước đó.
    - Các trọng số của kết nối giữa các neuron trong fully connected layer được học trong quá trình huấn luyện mô hình để tối ưu hóa hàm mất mát.
    - Fully connected layer thường được sử dụng trong các bài toán phân loại và dự đoán.

- **[27] Label function** là một hàm số trong machine learning được sử dụng để gán một nhãn (label) cho một mẫu dữ liệu.
    - Nó là một phần quan trọng của các thuật toán supervised learning, nơi dữ liệu huấn luyện đã được gán nhãn trước đó và mục tiêu của mô hình là học cách dự đoán đúng nhãn cho các dữ liệu mới dựa trên những gì đã học được từ dữ liệu huấn luyện.
- **[28]** Trong Machine Learning, **weight function** (hàm trọng số) là một hàm số được sử dụng để tính toán các trọng số (weights) cho các kết nối giữa các neuron trong mạng neural.
    - Các trọng số này đóng vai trò quan trọng trong việc học và dự đoán đầu ra của mô hình.
    - Trọng số càng chính xác thì kết quả dự đoán của mô hình sẽ càng tốt.
- **[29] Kỹ thuật attention** là một phương pháp trong lĩnh vực học sâu (deep learning) giúp mô hình tập trung vào các phần quan trọng của dữ liệu đầu vào. Nó cho phép mô hình xác định những vị trí quan trọng, đóng góp nhiều vào kết quả dự đoán cuối cùng.
    - Trong ngữ cảnh của mạng nơ-ron học sâu, kỹ thuật attention cho phép mô hình "chú ý" vào các phần của dữ liệu có ý nghĩa hoặc quan trọng hơn đối với quá trình dự đoán. Thay vì xem xét toàn bộ dữ liệu đầu vào một cách đồng đều, attention tập trung vào các phần tử quan trọng bằng cách thực hiện trọng số hoặc điểm số cho từng phần tử trong quá trình tính toán.
    - Phương pháp attention thường áp dụng cho các tác vụ như dịch máy (machine translation), nhận dạng hình ảnh (image recognition), nhận dạng giọng nói (speech recognition), và xử lý ngôn ngữ tự nhiên (natural language processing). Nó giúp mô hình tập trung vào các thông tin quan trọng trong dữ liệu đầu vào và thực hiện các phép toán dựa trên những phần tử quan trọng này.
- **[30] Data augmentation** là một kỹ thuật trong lĩnh vực Machine Learning (Học Máy) để mở rộng và tạo ra thêm dữ liệu huấn luyện từ dữ liệu huấn luyện ban đầu. Ý tưởng chính của data augmentation là thay đổi hoặc biến đổi dữ liệu huấn luyện bằng cách áp dụng các phép biến đổi nhỏ như xoay, thu phóng, cắt, lật, thay đổi màu sắc, hoặc thêm nhiễu vào dữ liệu gốc.
    - Mục tiêu của data augmentation là tăng cường độ đa dạng và phong phú của dữ liệu huấn luyện, từ đó giúp mô hình học máy học được các đặc trưng chung và chống lại overfitting (quá khớp). Khi áp dụng data augmentation, số lượng mẫu huấn luyện được tăng lên mà không cần thu thập thêm dữ liệu mới.
    - Ví dụ, trong tác vụ phân loại ảnh, data augmentation có thể bao gồm việc xoay, thu phóng, lật ảnh theo các góc độ khác nhau, thay đổi ánh sáng, hay áp dụng các biến đổi ngẫu nhiên khác để tạo ra các phiên bản khác nhau của cùng một hình ảnh.

- **[31]** *Aleesa, A., Younis, M. O. H. A. M. M. E. D., Mohammed, A. A., & Sahar, N. (2021). Deep-intrusion detection system with enhanced UNSW-NB15 dataset based on deep learning techniques. Journal of Engineering Science and Technology, 16(1), 711-727.*
  - http://jestec.taylors.edu.my/Vol%2016%20issue%201%20February%202021/16_1_49.pdf
- **[32]** *Meftah, S., Rachidi, T., & Assem, N. (2019). Network based intrusion detection using the UNSW-NB15 dataset. International Journal of Computing and Digital Systems, 8(5), 478-487.*
  - https://journal.uob.edu.bh/handle/123456789/3580
- **[33]** *Smys, S., Basar, A., & Wang, H. (2020). Hybrid intrusion detection system for internet of things (IoT). Journal of ISMAC, 2(04), 190-199.*
  - https://doi.org/10.36548/jismac.2020.4.002

HẾT