# Mục lục

# Thuật ngữ

**Artificial Intelligence** – Trí tuệ nhân tạo

**Bias** – Độ lệch giá trị trung bình

**Convolutional Neural Network** – Mạng nơ-ron tích chập

**Intrusion Detection System** – Hệ thống phát hiện xâm nhập

**Grayscale** – Thước xám

**Neural Network** – Mạng nơ-ron

**Overfitting** – Quá khớp

**ReLU – Activation Function** – Hàm kích hoạt

**Raw Data** – Dữ liệu thô

**Render** – Kết xuất

**Stride** – Bước nhảy

**Weight** – Trọng số

**Federated Learning / Collaborative Learning** – Học cộng tác

# Danh mục hình vẽ

**Hình 1** – Kiến trúc khái quát của mạng nơ-ron. [2]

**Hình 2** – Kiến trúc của CNN model (VGG-16)[5]

# Danh mục bảng

# Nghiên cứu cơ sở lý thuyết

## 1. Mạng nơ-ron – Neural Network [1][2]

Là phương thức trong trí tuệ nhân tạo dùng để huấn luyện máy tính xử lý và tính toán dữ liệu thô. Sử dụng các nơ-ron hoặc node tạo thành cấu trúc phân lớp có tính thích ứng cao từ đó liên tục cải thiện để cho ra kết quả tốt nhất. Được sử dụng để giải quyết những vấn đề phức tạp như dự đoán, nhận diện, …

Một kiến trúc mạng nơ-ron sẽ bao gồm ba loại layer :

* Input Layer : Là dữ liệu đầu vào được cho vào model. Số lượng phần tử nơ-ron ở layer này đại diện cho số lượng feature của dữ liệu.
* Hidden Layer(s) : Dữ liệu đầu vào sẽ được đưa từ Input Layer sang các Hidden Layer. Từng layer có thể có số lượng nơ-ron khác nhau. Đây là nơi output từ layer trước sẽ được nhân với trọng số (weight) của layer đó, tiếp đó là độ lệch giá trị trung bình (bias) và hàm kích hoạt (Relu).
* Output Layer : Là nơi dữ liệu đã được xử lý đi ra. Tương tự với Input Layer, chỉ có duy nhất một output layer duy nhất với nhiều phần tử nơ-ron.
* 
* Hình 1 – Kiến trúc khái quát của mạng nơ-ron

## 2. Mạng nơ-ron tích chập - Convolutional Neural Network[3][4]

Là một dạng của mạng nơ-ron bao gồm nhiều lớp nơ-ron mà trong đó có ít nhất một lớp tích chập mang cho mình nhiệm vụ xử lý dữ liệu được đưa vào. Một mạng nơ-ron tích chập sẽ bao gồm những layer như sau :

* Input Layer : Như Input Layer của mạng nơ-ron, chỉ có một Input Layer duy nhất là nơi dữ liệu thô được đưa vào.
* Convolution Layer(s) : Bao gồm các bộ lọc. Ở mỗi nơ-ron tích chập, dữ liệu đầu ra có chiều dài và độ cao nhỏ hơn so với dữ liệu thô nhưng cùng độ sâu (Ví dụ như dữ liệu thô là ảnh định dạng RGB thì độ sâu sẽ là 3 cho dữ liệu đầu ra). Vì thế nên khi dữ liệu thô đi qua mỗi lớp (layer) tích chập ta sẽ có dữ liệu đầu ra có cùng chiều dài và độ cao, độ sâu của dữ liệu đầu ra sẽ được quyết định bởi số bộ lọc và bước nhảy ở lớp tích chập.
* Activation Function Layer : Dữ liệu từ lớp tính chập đi qua đây sẽ được áp dụng các hàm tính phi tuyến tính như ReLU, Sigmoid, Tanh, …
* Pool Layer : Được đưa vào mạng nơ-ron tính chập nhằm giảm kích thước của khối nhằm tăng khả năng tính toán cho những lớp sau và đồng thời tránh vấn đề quá khớp. Thường có hai loại là max-pooling và average-pooling (hay còn gọi là mean pooling).
* Fully-connected Layer : Kế thừa dữ liệu từ các lớp trước, mỗi nơ-ron sẽ được kết nối với toàn bộ các nơ-ron trước. Thường nằm ở cuối mạng tính toán các điểm số (score) của lớp như độ chính xác của từng lớp.

## 3. Máy học cộng tác - Federated Learning[6][7][8]

## 4. Intrusion Detection System – Hệ thống phát hiện xâm nhập

## 5.

# Khảo sát các nghiên cứu liên quan

## Model cho mạng nơ-ron tích chập

Dựa trên một số nghiên cứu về hệ thống phát hiện xâm nhập trên nền tảng học sâu, chúng tôi sẽ sử dụng VGG-16 – Một model mạng nơ-ron tích chập sâu.

VGG-16 được Karen Simonyan và Andrew Zisserman giới thiệu vào 2014 ở cuộc thi ILSVRC 2014 (<https://www.image-net.org/challenges/LSVRC/2014/>) qua bài báo (<https://arxiv.org/abs/1409.1556>)

Về kiến trúc của VGG-16, Input dành cho model này sẽ là một ảnh RGB size 224x224. Sẽ có tất cả 16 lớp, 5 lớp tích chập được theo sau bởi max pooling. Tất cả các lớp tích chập sẽ sử dụng bộ lọc có kích thước 3x3 (hoặc 1x1). Thứ tự các lớp được thể hiện ở Hình 1.



Hình 2 – Kiến trúc của CNN model (VGG-16)

## Tập dữ liệu ([The UNSW-NB15 Dataset | UNSW Research](https://research.unsw.edu.au/projects/unsw-nb15-dataset))

Tập dữ liệu UNSW-NB15 được tạo bởi UNSW Canberra nhằm mô tả hoạt động mạng hỗn hợp bao gồm hoạt động bình thường và các hành vi tấn công mạng. Tập dữ liệu bao gồm 9 phân loại tấn công và 49 đặc trưng. Số lượng bản ghi của mỗi tập dữ liệu con và tập dữ liệu mẹ được thể hiện qua bảng sau

|  |  |  |
| --- | --- | --- |
| Tên lớp | Tập dữ liệu con dành cho huấn luyện | Tập dữ liệu con dành cho thử |
| Normal | 56,000 | 37,000 |
| Generic | 40,000 | 18,871 |
| Exploits | 33,393 | 11,132 |
| Fuzzers | 18,184 | 6062 |
| DoS | 12,264 | 4089 |
| Reconnaissance | 10,491 | 3496 |
| Analysis | 2000 | 677 |
| Backdoor | 1746 | 583 |
| Shellcode | 1133 | 378 |
| Worms | 130 | 44 |
| Tổng cộng | 175,343 | 82,337 |

Bảng 1 – Số lượng bản ghi của tập dữ liệu huấn luyện và thử

## Tiền xử lý dữ liệu ([2103.07765.pdf (arxiv.org)](https://arxiv.org/ftp/arxiv/papers/2103/2103.07765.pdf))

### 1. Cho DCNN

Sử dụng mã hóa one-hot, ta có thể biến đổi các bản ghi của tập dữ liệu sang loại dữ liệu đầu vào phù hợp với mạng nơ-ron tích chập.

Đầu tiên tập dữ liệu gốc sẽ được tách từng hàng ngang (mỗi hàng ngang tương đương với một bản ghi). Mỗi bảng ghi đã được tách ra sẽ chuyển về dạng bảng với mỗi ô là một đặc tính, từ đó tạo nên bảng vuông có độ dài là 16 đặc tính tương đương với 16 ô. Những ô không chứa dữ liệu của bất kỳ đặc tính nào sẽ được pad để đảm bảo kích thích của ảnh sau này.

Table

Description automatically generated

Hình 3 – Cấu trúc của một bản ghi mẫu đã được chuyển sang dạng 16x16

Sau đó mã hóa one-hot sẽ đưa giá trị của các đặc trưng từ giá trị dạng số trở thành giá trị thước xám. Với các ô không chứa dữ liệu (pad) sẽ được biến đổi thành các pixel màu đen (có giá trị bằng 255). Sau cùng thì tất cả các đặc trưng (dịch vụ, giao thức, trạng thái, …) của bản ghi sẽ được biến đổi thành 1 nếu như có xuất hiện.

Số lượng bản ghi sẽ được giữ nguyên như ở Hình 1.

## Phương pháp đánh giá hiệu quả

### 1. Cho DCNN

## Mô hình triển khai học cộng tác

# Thực nghiệm

## 1. DCNN

Về tập dữ liệu thực nghiệm, nhóm sử dụng tập dữ liệu UNSW-NB15 đã qua quá trình tiền xử lý nhằm cho ra dữ liệu đầu vào phù hợp với DCNN.

### 1. Local model và Ideal model

Ở Local model, model VGG-16 sẽ được huấn luyện ngay chính trên thiết bị sinh dữ liệu. Đối với Ideal model, dữ liệu trên nhiều thiết bị sinh ra sẽ được tập hợp lại ở một máy trung tâm cho việc huấn luyện.

### 2. Federated Learning model

# References

[1] : [What is a Neural Network? AI and ML Guide | AWS](https://aws.amazon.com/what-is/neural-network/)

[2] : [What are Neural Networks? | IBM](https://www.ibm.com/cloud/learn/neural-networks)

[3] : [Introduction to Convolution Neural Network | GeeksforGeeks](https://www.geeksforgeeks.org/introduction-convolution-neural-network/)

[4] : Deep Learning (Ian J. Goodfellow, Yoshua Bengio and Aaron Courville), MIT Press, 2016.

[5] : [Using Deep Learning Models / Convolutional Neural Networks | ecognition.com](https://docs.ecognition.com/eCognition_documentation/User%20Guide%20Developer/8%20Classification%20-%20Deep%20Learning.htm)

[6] : [Federated Learning: Collaborative Machine Learning without Centralized Training Data | Google AI Blog](https://ai.googleblog.com/2017/04/federated-learning-collaborative.html)

[7] : [What is federated learning? | IBM Research Blog](https://research.ibm.com/blog/what-is-federated-learning)

[8] : [What is Federated Learning? | Google Cloud Tech](https://www.youtube.com/watch?v=X8YYWunttOY)

[9] :

[10] :

[11] :

[12] :

[13] :

[14] :

[15] :

[16] :

[17] :

[18] :

[19] :

[20] :

[21] :

[22] :