TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN – ĐHQG-HCM
Khoa Mạng máy tính & Truyền thông

Tổng quan: Học máy cho IDS

NT204 – Hệ thống tìm kiếm, phát hiện và ngăn ngừa xâm nhập

GV: Đỗ Hoàng Hiển

hiendh@uit.edu.vn

Nội dung hôm nay…

IDS dựa trên học máy

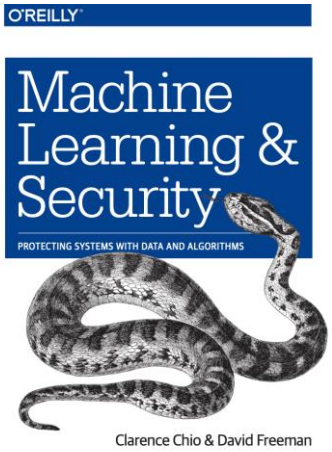
Hôm nay có gì?

Các phương pháp học máy cho IDS

- Tổng quan về nghiên cứu gần đây

Tài liệu tham khảo:

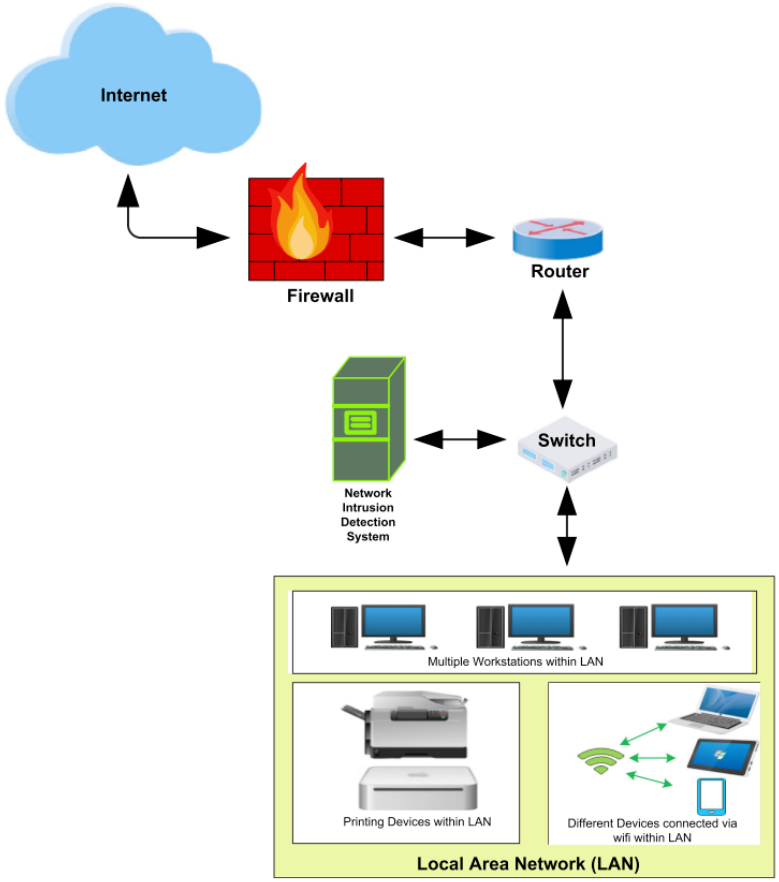
1. Chio, C., & Freeman, D. (2018). *Machine Learning & Security* book
2. Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J., & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. Transactions on Emerging Telecommunications Technologies, 32(1), 1–29.
3. Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity, 2(1)



Những kiến thức đã học…

Nhắc lại

- **Intrusion – xâm nhập:** hành động truy cập trái phép vào thông tin bên trong 1 máy tính hoặc hệ thống mạng, ảnh hưởng đến tính bảo mật, toàn vẹn, và sẵn sàng của nó
- **Intrusion detection and prevention systems** (IDPS) thường chủ yếu tập trung *xác định các xâm nhập, ghi log các thông tin, cố gắng ngăn chặn và báo cáo với các quản trị viên bảo mật*
  - **Mục tiêu:** đảm bảo an toàn thông tin cho một mạng hoặc hệ thống máy tính theo **bộ ba CIA**

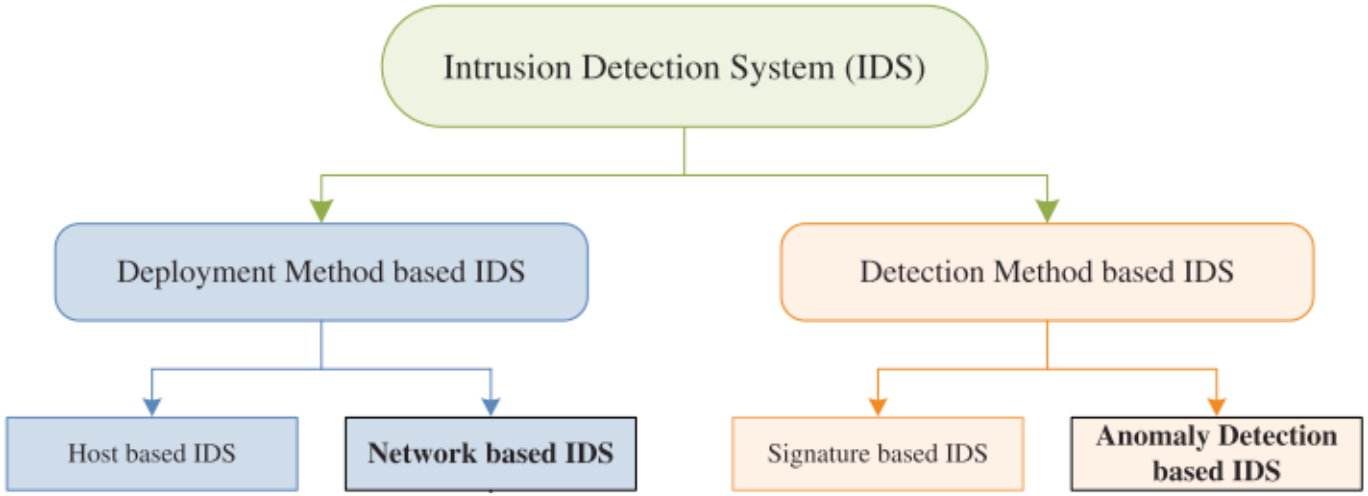


Một NIDS được triển khai mở mode Passive [1]

Các loại IDS

Phân loại IDS

- **Dựa trên kỹ thuật phát hiện**
  - Signature-based (SIDS)
  - Anomaly-based (AIDS)
- **Dựa trên cách triển khai/nguồn dữ liệu**
  - Network-based (NIDS)
  - Host-based (HIDS)



Các thách thức với IDS truyền thống

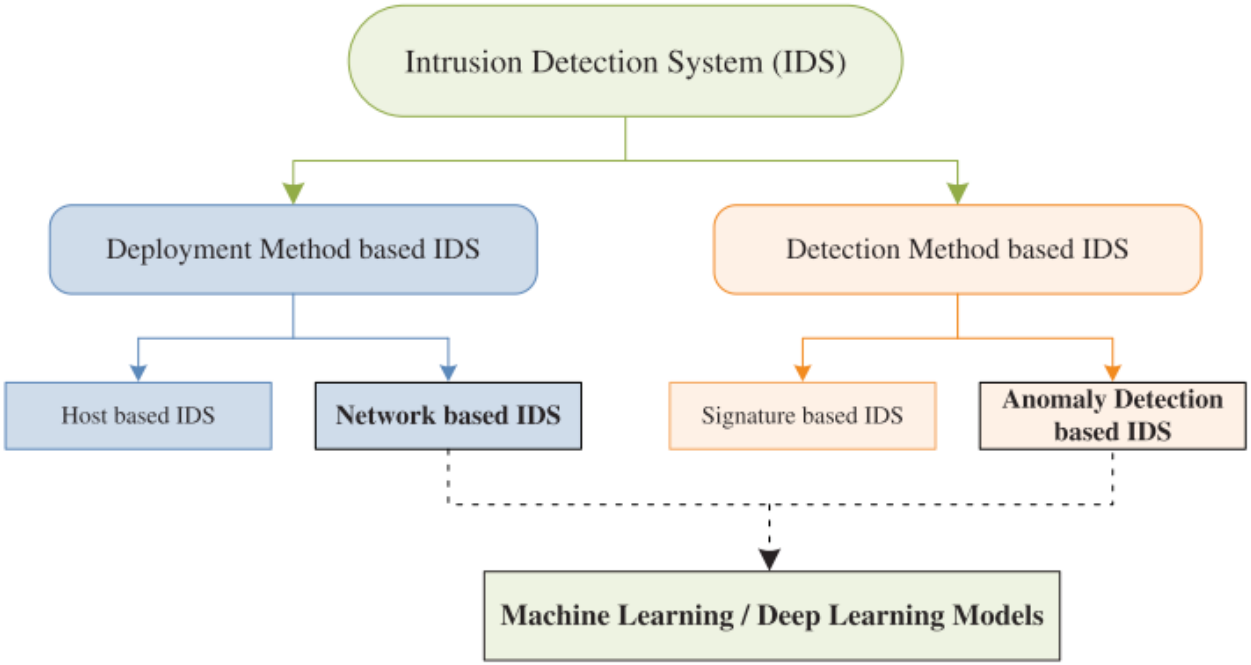
- **Kích thước mạng** và **dữ liệu liên quan** ngày càng tăng
- Thách thức trong việc cải thiện **độ chính xác trong phát hiện tấn công, đồng thời giảm tỉ lệ cảnh báo sai**
- Nhiều novel/zero-day attack→ **làm sao để phát hiện được?**
- Các tấn công mạng và các kỹ thuật qua mặt IDS ngày càng **phức tạp**

→ **Giải pháp tiềm năng:** IDS dựa trên Machine Learning và Deep Learning

Các loại IDS

Loại IDS mới

- **Dựa trên kỹ thuật phát hiện**
  - Signature-based (SIDS)
  - Anomaly-based (AIDS)
- **Dựa trên cách triển khai**
  - Network-based (NIDS)
  - Host-based (HIDS)



→ Chúng ta sẽ tập trung vào các phương pháp ML/DL khác nhau được dùng trong NIDS

Machine Learning – Học máy: Tổng quan

- Nhìn chung, ***ML (Học máy) là quá trình sử dụng các dữ liệu đã thấy để đưa ra thuật toán dự đoán cho những dữ liệu chưa từng thấy (dữ liệu tương lai)***
- Một thuật toán machine learning có:
  - Dữ liệu đầu vào là **tập dữ liệu huấn luyện *(training dataset)***
  - Kết quả đầu ra là 1 **mô hình *(model)***
  - **Model** là một thuật toán nhận đầu vào là các dữ liệu mới có cùng định dạng với dữ liệu huấn luyện và đưa ra một dự đoán

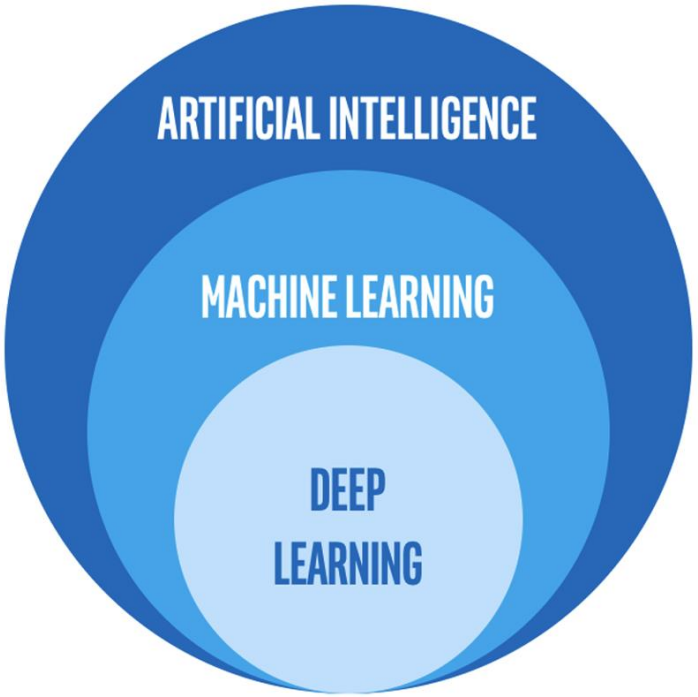


Photo: Intel

Supervised learning – Học có giám sát

- **Supervised learning – Học có giám sát**: các thông tin hữu ích sẽ được trích xuất từ các dữ liệu đã được gán nhãn (label)
  - **Classification – Phân lớp:** xác định thể loại của mỗi dữ liệu. Số lượng các lớp (class) có thể là:
    - ***Nhị phân:*** chỉ có 2 lớp. Ví dụ, phân loại tất cả các sự kiện mạng thành *tấn công* hoặc *bình thường*
    - ***Đa lớp***: Có nhiều lớp. Ví dụ, cần xác định cụ thể mỗi loại malware là ransomware, key- logger, hay remote access trojan
  - **Regression – Hồi quy**: cố gắng dự đoán 1 giá trị số thực
    - Ví dụ, dự đoán số lượng email lừa đảo 1 nhân viên sẽ nhận được trong vòng 1 tháng, dựa trên các thông tin về vị trí làm việc, quyền hạn, thời gian làm việc trong công ty, điểm đánh giá về bảo mật cá nhân…
    - **Kỹ thuật phát hiện *Anomaly-based*** là một điển hình của hồi quy: xác định khi nào một giá trị quan sát được đủ khác giá trị dự đoán được để xác định điều bất thường đang diễn ra trong hệ thống

Unsupervised learning – Học không giám sát

- **Unsupervised learning – Học không giám sát**: dùng các dữ liệu chưa được gán nhãn để trích xuất các đặc điểm và thông tin có ích
  - **Clustering**: xác định các dữ liệu nào tương tự nhau
    - Ví dụ, để phân tích một lượng lớn các dữ liệu traffic đến 1 website, có thể cần gom nhóm các request với nhau. Một số nhóm như: botnet, hoặc người dùng thông thường

Khác:

- **Semi-Supervised Learning – Học bán giám sát:** Dữ liệu đầu vào bao gồm cả dữ liệu đã gán nhãn và chưa gán nhãn
- **Reinforcement Learning – Học tăng cường:** đưa ra các dự đoán dựa trên việc thử và sai, dạy cho các máy (agent) thực hiện tốt 1 nhiệm vụ (task) bằng tương tác với môi trường (environment) thông qua hành động (action) và nhận được phần thưởng (reward)

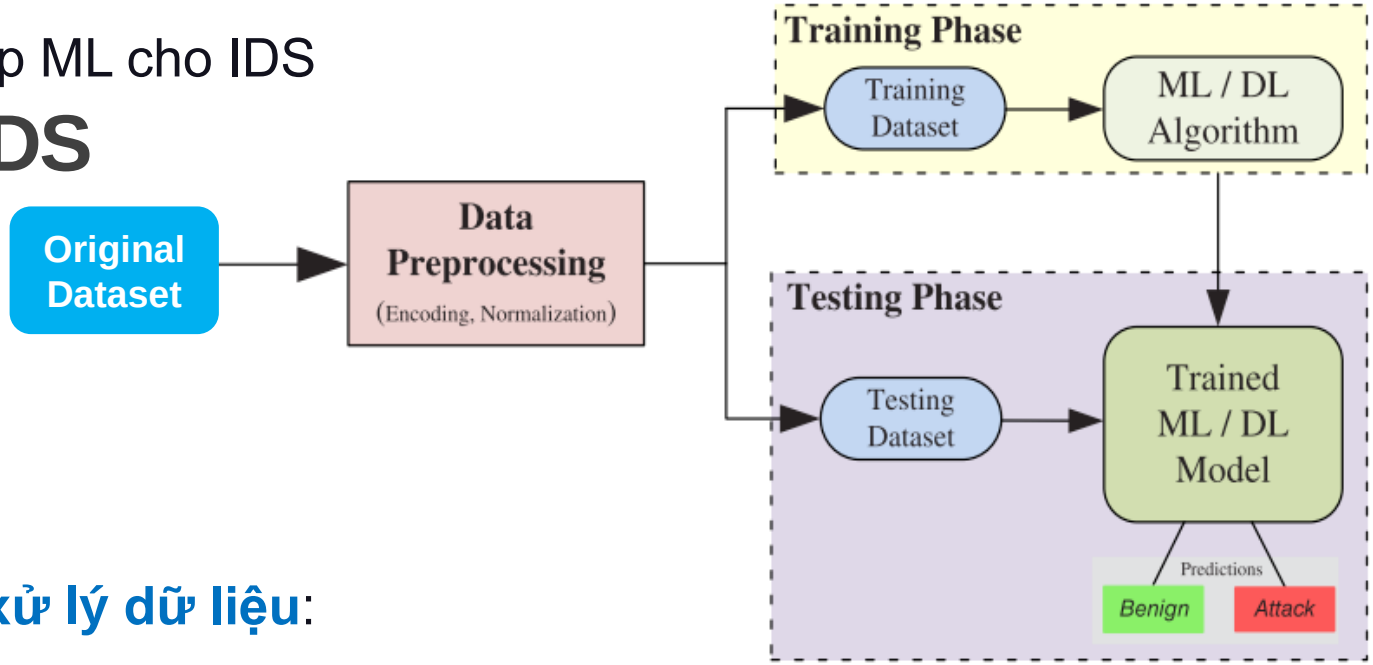
Read more: https://machinelearningmastery.com/a-tour-of-machine-learning-algorithms/

Bộ dữ liệu

#	Age	Has_Job	Own_House	Credit_Rating	Class
1	Young	false	false	fair	No
2	Young	false	false	good	No
3	Young	true	false	good	Yes
4	Young	true	true	fair	Yes
5	Young	false	false	fair	No
6	middle	false	false	fair	No
7	middle	false	false	good	No
8	middle	true	true	good	Yes
9	middle	false	true	excellent	Yes
10	middle	false	true	excellent	Yes
11	old	false	true	excellent	Yes
12	old	false	true	good	Yes
13	old	true	false	good	Yes
14	old	true	false	excellent	Yes
15	old	false	false	fair	No

Các phương pháp ML cho IDS

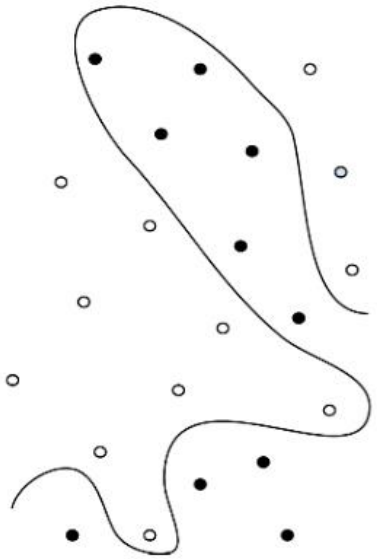
ML-based IDS



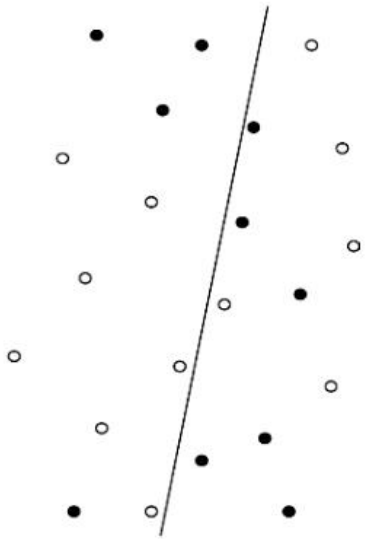
- **3 bước chính:**
  - (1) Bước **Tiền xử lý dữ liệu**:
    - *Chuyển dữ liệu sang định dạng phù hợp để dùng trong thuật toán*
    - *Thường bao gồm: Encoding, normalization, data cleaning*
    - *Chia ngẫu nhiên thành 2 tập dữ liệu **training** và **testing** (thường là 80%—20%)*
  - (2) Bước **Training – Huấn luyện**:
    - *Thuật toán ML hoặc DL được huấn luyện với tập dữ liệu **training***
  - (3) Bước **Testing – Kiểm tra**:
    - *Model đã huấn luyện được kiểm tra lại với tập dữ liệu **testing** và được đánh giá dựa trên các dự đoán mà nó đưa ra*

Overfitting và Underfitting

- **Overfitting:** model tạo ra quá khớp với dữ liệu huấn luyện nên không tổng quát hóa tốt các trường hợp dữ liệu chưa thấy (dữ liệu mới)
- **Underfitting:** model quá đơn giản cũng có thể tổng quát hóa kém các dữ liệu chưa thấy



Overfit



Underfit

"Shallow" Learning – Học "Cạn"

Các thuật toán Machine Learning

ML là 1 tập con của AI, gồm các phương pháp và thuật toán cho phép máy tính tự động học bằng cách sử dụng với **model toán học** để trích xuất các thông tin hữu ích từ các tập dữ liệu lớn

Các giải thuật ML phổ biến được dùng cho IDS:

- **Decision Tree (DT)**
- **K-Nearest Neighbor (KNN)**
- **Support Vector Machine (SVM)**
- **K-Means Clustering**
- **Ensemble Methods**

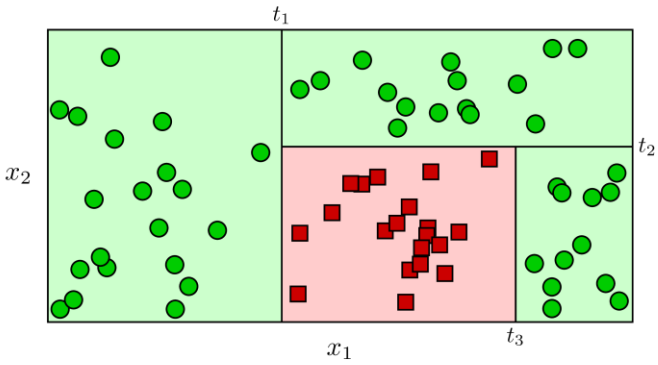
Thực hành: Sử dụng thư viện **sklearn**

- https://scikit-learn.org/
- https://scikit-learn.org/stable/auto_examples/index.html#general-examples

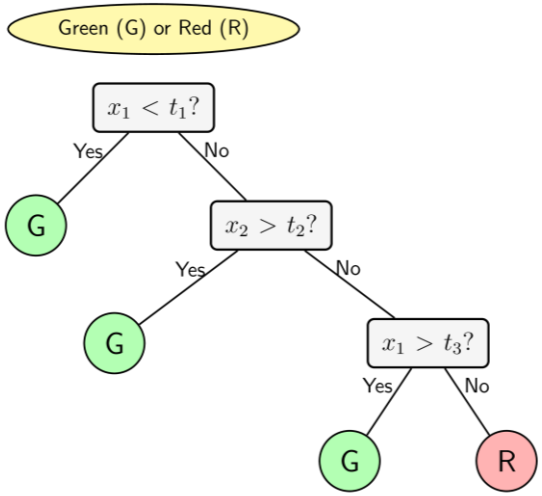
Decision Tree (DT) – Cây quyết định

- Thuật toán học máy có giám sát cơ bản, dùng cho cả 2 bài toán **classification** và **regression** trên tập dữ liệu cho trước bằng cách áp dụng một loạt các quyết định (rules)
  - Model có cấu trúc cây với các node, nhánh và lá (leaf)
  - Mỗi node đại diện cho 1 đặc điểm. Mỗi nhánh đại diện cho 1 quyết định hoặc 1 rule, mỗi lá là 1 kết quả hoặc nhãn cần gán cho dữ liệu

  → DT tự động chọn các thuộc tính tốt nhất để dựng cây và thực hiện cắt tỉa (pruning) cây, bỏ đi các nhánh không liên quan để tránh overfitting



(a)

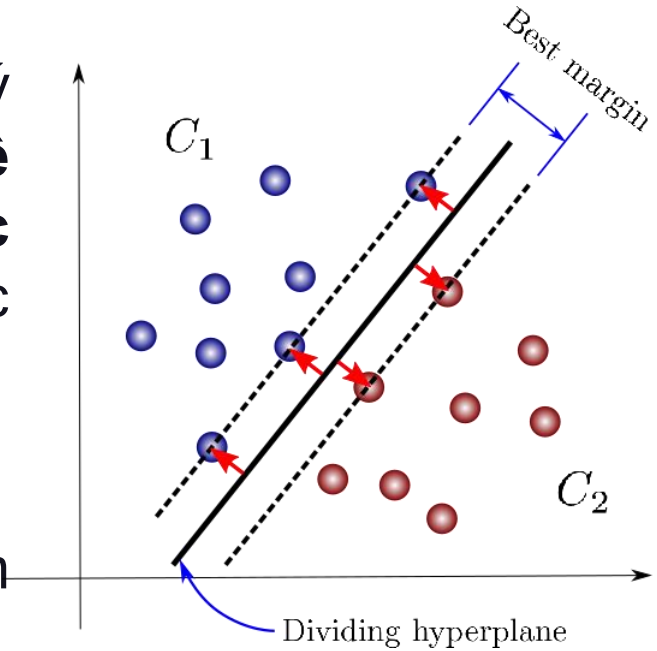


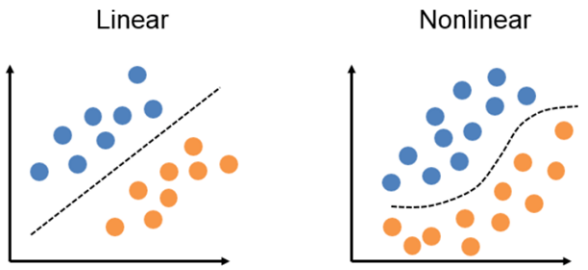
(b)

- **Các model DT phổ biến:** CART, C4.5, và ID3
- **Khác:** Random Forest (RF), XGBoost

Support vector machine (SVM)

- Thuật toán học có giám sát đơn giản với ý tưởng **siêu mặt phẳng phân chia có lề (margin) lớn nhất** trong **không gian thuộc tính n chiều**, dùng làm giải pháp cho các vấn đề tuyến tính và phi tuyến tính
  - n = 2: đường thẳng, n = 3: mặt phẳng
- Với các vấn đề phi tuyến tính, các hàm kernel (kernel function) được sử dụng
  - (1) ánh xạ 1 vector đầu vào ít chiều vào không gian thuộc tính nhiều chiều với hàm kernel.
  - (2) sử dụng các support vectors để thu được một mặt phẳng phân chia có lề lớn nhất, hoạt động như ranh giới để phân loại.

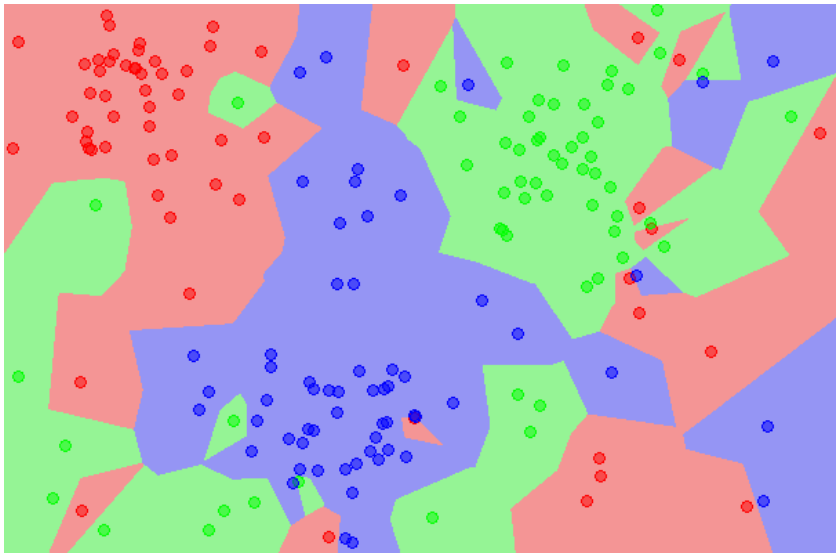




“Shallow” Learning – Học “Cạn”

K-Nearest Neighbor (KNN)

- Thuật toán học máy có giám sát đơn giản sử dụng ý tưởng **“thuộc tính giống nhau”** để dự đoán lớp phân loại của dữ liệu
  - Xác định dữ liệu dựa trên các “hàng xóm” gần giống bằng cách tính toán khoảng cách từ các hàng xóm này
    - k = số “hàng xóm” cần xem xét
  - Tham số k ảnh hưởng đến hiệu suất của model
    - **Quá nhỏ:** model dễ bị overfitting
    - **Quá lớn:** phân loại sai dữ liệu

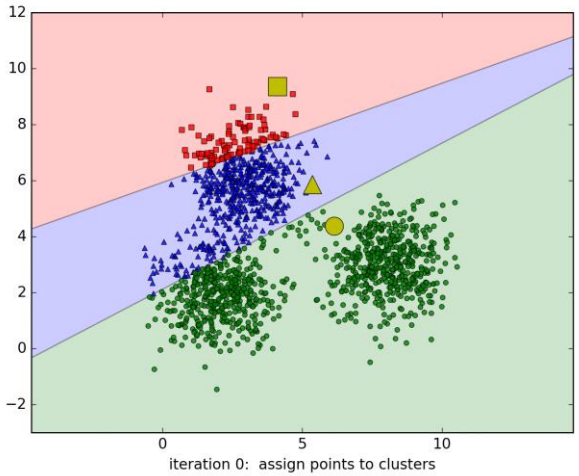


"Shallow" Learning – Học "Cạn"

K-mean clustering

- Thuật toán ML phổ biến, học theo dạng **không giám sát** với **việc lặp lại dựa trên điểm trung tâm**
- **Ý tưởng:** chia dữ liệu thành các clusters (hoặc nhóm) bằng cách đặt các dữ liệu giống nhau nhiều vào chung cluster
  - K là số trung tâm (của cluster/nhóm) có trong bộ dữ liệu
  - Thường tính toán distance – khoảng cách để có thể gán dữ liệu vào 1 cluster/nhóm

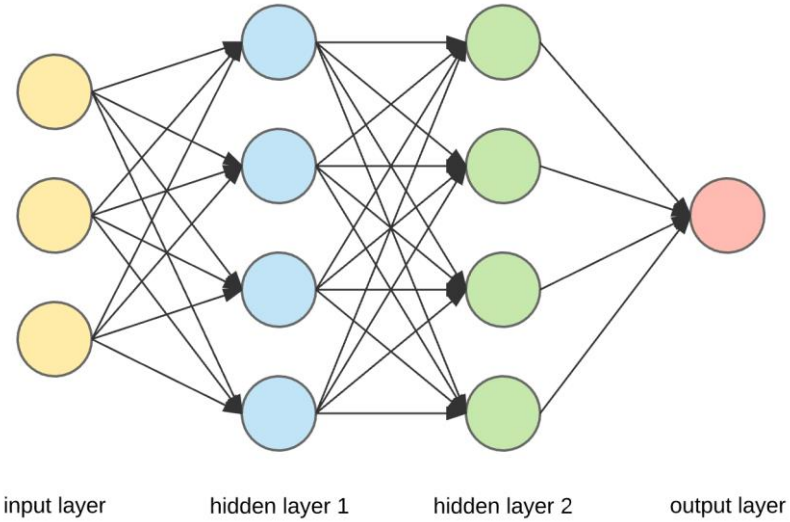
→ Ý tưởng chính là để giảm tổng khoảng cách giữa các dữ liệu và trung tâm đại diện của cluster/nhóm



"Shallow" Learning – Học "Cạn"

Artificial Neural Network (ANN)

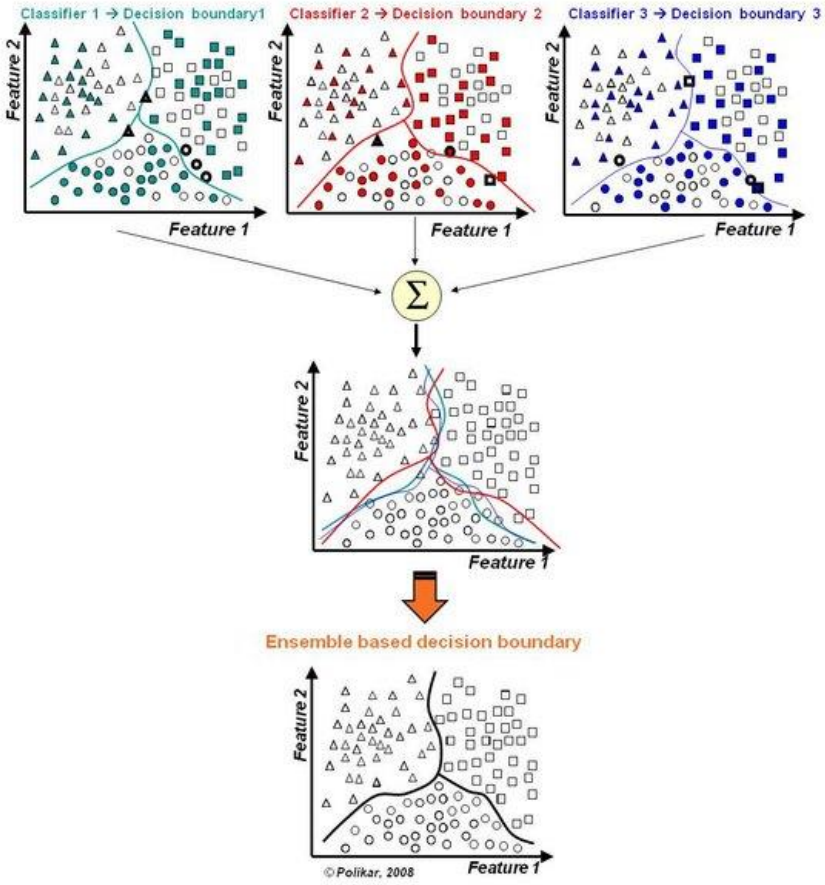
- **ANN** là một thuật toán học có giám sát lấy ý tưởng từ hoạt động của hệ thống thần kinh trong não bộ con người (mạng thần kinh sinh học)
  - Bao gồm nhiều thành phần xử lý là các *neuron (node)* và các kết nối giữa chúng
  - Các node được tổ chức thành 1 layer input, các layer ẩn, và 1 layer output
  - Thuật toán **backpropagation (lan truyền ngược)** là một kỹ thuật học trong ANN



Xem thêm: https://www.superdatascience.com/blogs/the-ultimate-guide-to-artificial-neural-networks-ann
Demo: https://playground.tensorflow.org/

Các phương pháp kết hợp

- **Ý tưởng chính:** tận dụng nhiều classifier khác nhau để học bằng cách kết hợp
- Cách học kết hợp nhằm kết hợp các classifier còn yếu bằng cách huấn luyện nhiều classifier để tạo thành 1 classifier tốt hơn thông qua việc lựa chọn 1 **thuật toán bầu chọn (voting)**



Neural network-based – Dựa trên mạng nơron

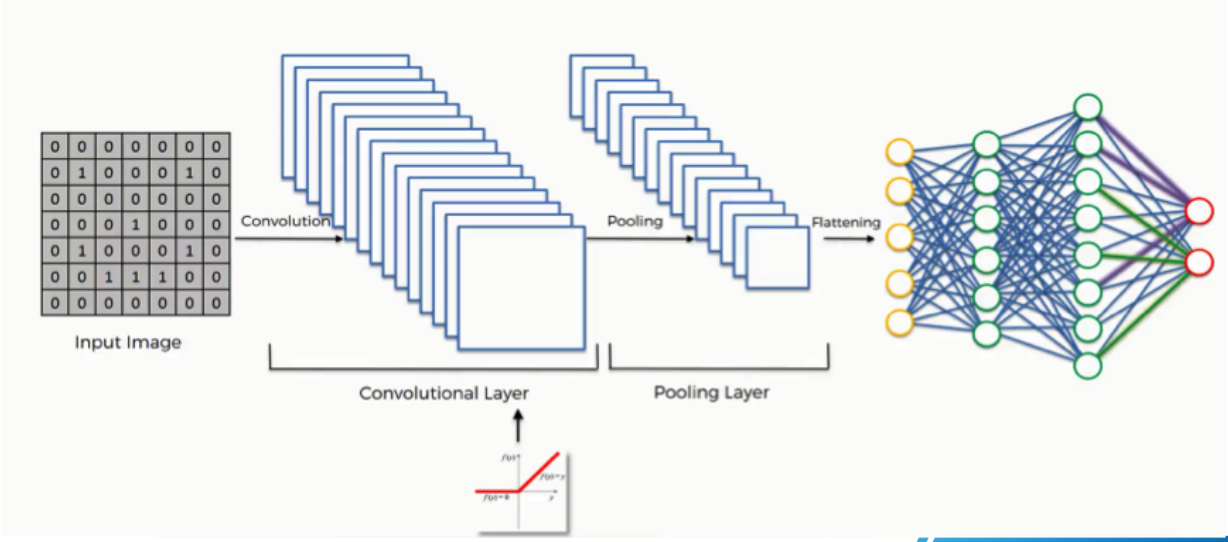
Các thuật toán Deep Learning (DL)

- DL là 1 tập con của ML gồm nhiều **hidden layers** để có đặc điểm của deep neural network
- Những kỹ thuật này hiệu quả hơn so với ML nhờ vào **cấu trúc sâu** (deep structure) và khả năng tự học các thuộc tính quan trọng từ bộ dữ liệu và tạo kết quả đầu ra
- Các thuật toán DL phổ biến dùng cho IDS:
  - **Convolutional neural network (CNN)**
  - **Recurrent Neural Networks (RNN)**
  - **AutoEncoder (AE)**
  - **Deep belief network (DBN)**

Các thuật toán DL

Convolutional Neural Network (CNN)

- Cấu trúc Deep learning phù hợp hơn với các dữ liệu lưu ở dạng mảng và CNN là thuật toán đã được sử dụng thành công trong lĩnh vực thị giác máy tính
- Gồm 1 **input layer**, chồng các layer **convolutional *(tích chập)*** và **pooling *(tổng hợp)*** để trích xuất thuộc tính, 1 layer **fully connected *(kết nối đầy đủ)*** và 1 layer **phân loại**
- Trong IDS, được dùng cho mục đích trích xuất thuộc tính và phân loại theo hướng có giám sát



Xem thêm:
https://www.superdatascience.com/blogs/the-ultimate-guide-to-convolutional-neural-networks-cnn

Các thuật toán DL

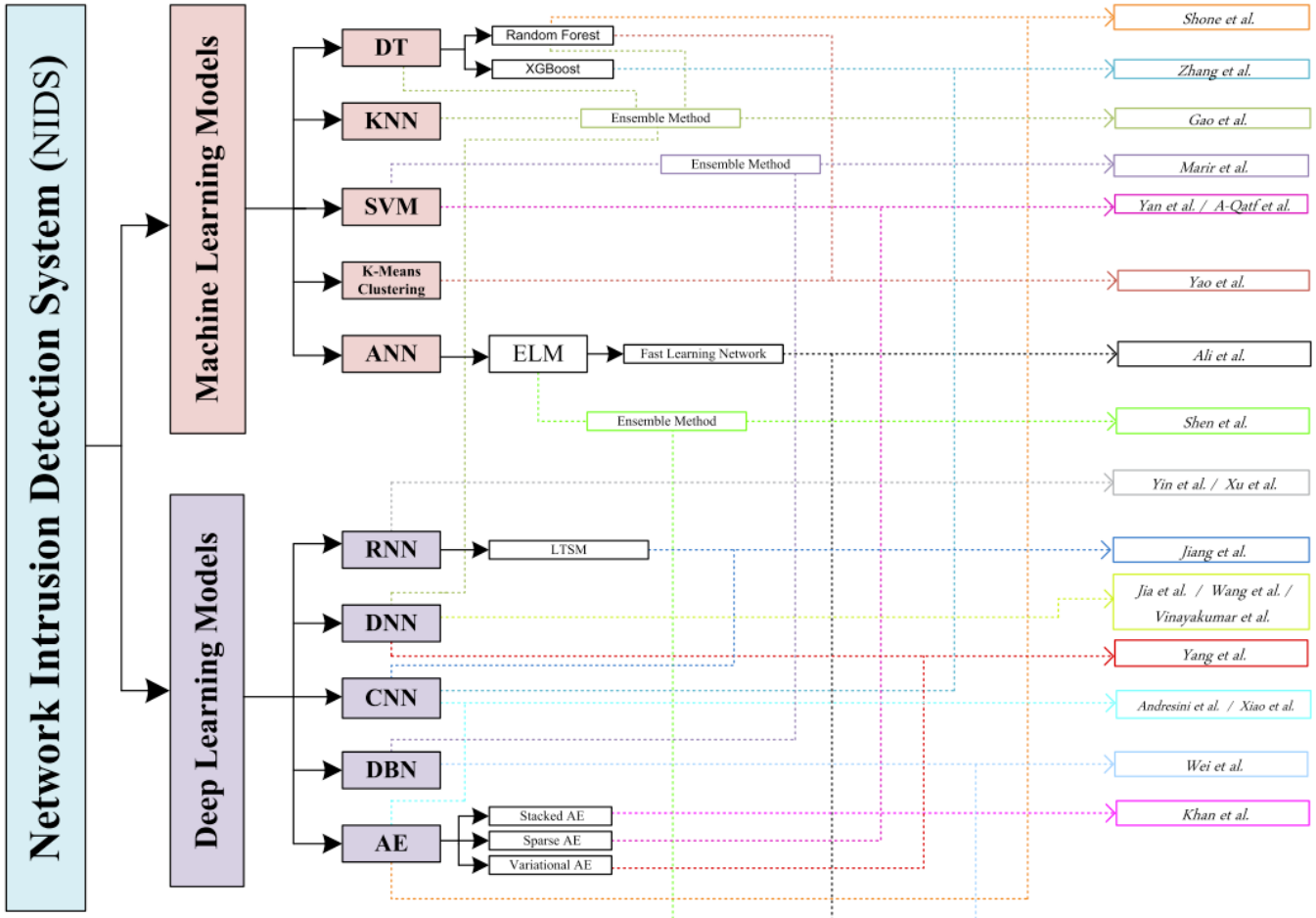
Recurrent Neural Networks (RNN)

- Recurrent Neural Networks (RNN) mở rộng khả năng của mạng neural feed-forward truyền thống và được thiết kế để xử lý **dữ liệu trình tự *(sequence data)***
  - RNN gồm các đơn vị (unit) input, hidden và output, các đơn vị ẩn là các thành phần bộ nhớ
  - Để đưa ra quyết định, mỗi đơn vị RNN dựa trên input hiện tại của nó và kết quả đầu ra của đầu vào trước đó
- Trong IDS, RNN có thể dùng để **phân loại và trích xuất thuộc tính theo dạng có giám sát.** RNN thường có thể xử lý các chuỗi trình tự có độ dài giới hạn, nếu độ dài quá lớn có thể bị ảnh hưởng do hạn chế bộ nhớ
- Một số biến thể của RNN: **Long short-term memory (LSTM), Gated Recurrent Unit (GRU)**

Read more: https://www.superdatascience.com/blogs/the-ultimate-guide-to-recurrent-neural-networks-rnn/

IDS dựa trên ML và DL

Tổng quan các loại thuật toán cho IDS



Read more: Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J., & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Transactions on Emerging Telecommunications Technologies*, *32*(1), 1–29. https://doi.org/10.1002/ett.4150

Làm sao để học ML?

Các tài liệu về ML

Một số tài liệu để tự học về ML/DL:

- **Machine Learning:**
  - Coursera - Machine Learning: https://www.coursera.org/learn/machine-learning
  - Udemy - Machine Learning A-Z: https://www.udemy.com/course/machinelearning/
- **Deep Learning:**
  - Coursera - Deep Learning Specialization: https://www.coursera.org/specializations/deep-learning
  - Udemy - Deep Learning A-Z: https://www.udemy.com/course/deeplearning
  - Book: Dive into Deep Learning - https://d2l.ai/ (EN) - https://d2l.aivivn.com/ (VN)
- Tài liệu tiếng Việt:
  - Machine Learning cơ bản: https://machinelearningcoban.com/
  - NTT - Deep Learning cơ bản: https://nttuan8.com/

Đánh giá IDS

Các tập dữ liệu benchmark

- **KDD Cup'99**
- **NSL-KDD**
- **Kyoto 2006+**
- **UNSW-NB15**
- **CIC-IDS2017**
- **CSE-CIC-IDS2018**

Dataset	Year	Attack types	Attacks
KDD Cup'99[129]	1998	4	DoS, Probe, R2L, U2R
Kyoto 2006+[130]	2006	2	Known Attacks, Unknown Attacks
NSL-KDD[131]	2009	4	DoS, Probe, R2L, U2R
UNSW-NB15[132]	2015	9	Backdoors, DoS, Exploits, Fuzzers, Generic, Port scans, Reconnaissance, Shellcode, worms
CIC-IDS2017[133]	2017	7	Brute Force, HeartBleed, Botnet, DoS, DDoS, Web , Infiltration
CSE-CIC-IDS2018[133]	2018	7	HeartBleed, DoS, Botnet, DDoS, Brute Force, Infiltration, Web.

Các tập dữ liệu benchmark (tt)

- **KDD Cup'99 (lỗi thời)**
  - Tập dữ liệu phổ biến nhất và được sử dụng rộng rãi cho IDS
  - Gồm **5 lớp phân loại** và gần **2 triệu records** cho huấn luyện và kiểm tra
  - Mỗi record gồm **41 thuộc tính khác nhau** và được gán nhãn là **normal** *(bình thường)* hoặc **attack** (*tấn công)*
  - Các tấn công được chia làm 4 loại khác nhau gồm **Denial of Service (DoS), Probe, Remote to Local (R2L),** và **User to Root (U2R)**
- **NSL-KDD (2009):**
  - Là phiên bản sửa đổi và tinh chỉnh của tập KDD Cup'99 bằng cách loại bỏ 1 số vấn đề tồn tại
  - Cũng là tập dữ liệu với các record gồm 41 thuộc tính, các tấn công cũng được chia làm 4 loại như trong KDD Cup'99
  - Chi tiết: https://www.unb.ca/cic/datasets/nsl.html

Các tập dữ liệu benchmark (tt)

- **Kyoto 2006+**
  - Được tạo từ các bản ghi traffic mạng thu được bằng cách triển khai các honeypot, sensors, server email, web crawler, và các biện pháp an ninh mạng khác của Đại học Kyoto, Nhật Bản
  - Tập dữ liệu mới nhất có các bản ghi traffic từ 2006 đến 2015
  - Mỗi bản ghi có **24 thuộc tính thống kê**, gồm 14 thuộc tính lấy từ tập KDD Cup'99 và 10 thuộc tính bổ sung
  - Chi tiết: http://www.takakura.com/Kyoto_data/

Các tập dữ liệu benchmark (tt)

- **UNSW-NB15**
  - Được tạo bởi Trung tâm An ninh mạng Úc (Australian Center for Cyber Security) và Đại học UNSW, Úc
  - Gồm gần 2 triệu bản ghi với **49 thuộc tính** được trích xuất từ Bro-IDS, công cụ Argus, và một số thuật toán mới
  - Gồm các loại tấn công: Worms, Shellcode, Reconnaissance, Port Scans, Generic, Backdoor, DoS, Exploits, và Fuzzers
  - Chi tiết: https://research.unsw.edu.au/projects/unsw-nb15-data-set

Các bộ dữ liệu benchmark (tt)

- **CIC-IDS2017**
  - Được tạo ra bởi Viện An ninh mạng Canada (Canadian Institute of Cyber Security - CIC) vào năm 2017
  - Mỗi record gồm **hơn 80 thuộc tính**
  - Gồm cả dữ liệu traffic bình thường và các tấn công cập nhật theo thực tế. Traffic mạng được phân tích bằng công cụ **CICFlowMeter** sử dụng các thông tin timestamps, địa chỉ IP nguồn và đích, giao thức và các tấn công
  - CICIDS2017 có các kịch bản tấn công phổ biến như Brute Force Attack, Heart Bleed Attack, Botnet, Denial of Service (DoS) Attack, Distributed DoS (DDoS) Attack, Web Attack, và Infiltration Attack
  - Chi tiết: https://www.unb.ca/cic/datasets/ids-2017.html

Các bộ dữ liệu benchmark (tt)

- **CSE-CIC-IDS2018**
  - Được tạo ra bởi Communications Security Establishment (CSE) kết hợp với CIC vào năm 2018
  - Có các profile người dùng với các sự kiện khác nhau
  - Để tạo tập dữ liệu, tất cả các profile được kết hợp với một tập các thuộc tính duy nhất
  - Mỗi record gồm **hơn 80 thuộc tính** khác nhau
  - Gồm 7 kịch bản tấn công khác nhau: Brute-force, Heartbleed, Botnet, DoS, DDoS, Web attacks, và xâm nhập mạng từ bên trong
  - Chi tiết: https://www.unb.ca/cic/datasets/ids-2018.html

Nhắc lại

Các chỉ số đánh giá IDS

- Các cảnh báo có thể được phân loại thành các nhóm sau:
  - **True Positive (TP)**: Các cảnh báo đã được xác nhận thực tế đúng là 1 tấn công
  - **False Positive (FP)**: Các cảnh báo nhưng thực tế không phải là 1 tấn công
  - **True Negative (TN)**: Không có tấn công nào xảy ra
  - **False Negative (FN)**: Một tấn công nhưng không bị phát hiện

		Dự đoán	
		Attack	Normal
Thực tế	Attack	**TP**	**FN**
	Normal	**FP**	**TN**

Nhãn nhị phân

		Dự đoán		
		a	b	c
Thực tế	a	**TP**	**FN**	**FN**
	b	**FP**	**TN**	**TN**
	c	**FP**	**TN**	**TN**

Đa nhãn

Đánh giá IDS

Các chỉ số đánh giá

- **Accuracy – Độ chính xác:** có bao nhiều trường hợp được xác định đúng (là tấn công hoặc bình thường) trong tổng số các trường hợp

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}.$$

- **False positive rate (FPR) và false negative rate (FNR):**
  - FPR xác định tỉ lệ trường hợp bình thường nhưng bị cảnh báo là tấn công
  - FNR xác định tỉ lệ trường hợp tấn công nhưng không được cảnh báo

$$FPR = \frac{FP}{FP + TN}, \quad FNR = \frac{FN}{FN + TP}.$$

Wu C, Li W. Enhancing intrusion detection with feature selection and neural network. Int J Intell Syst. 2021;1-19.

Các chỉ số đánh giá (tt)

- Để đánh giá chất lượng của một giải pháp phát hiện, chúng ta dựa trên tổ hợp 3 chỉ số đánh giá khác nhau: Recall, Precision và F1-score
- **Precision:** tỷ lệ các phát hiện chính xác trên tổng số các cảnh báo tấn công IDPS đã tạo ra

$$Precision = \frac{TP}{TP + FP}$$

- **Recall** *(R - detection rate):* tỷ lệ các phát hiện chính xác trên tổng số tất cả các trường hợp tấn công thực tế đã có

$$Recall = \frac{TP}{TP + FN}$$

- **F1-score:** tổ hợp 2 chỉ số *Precision* và *Recall*

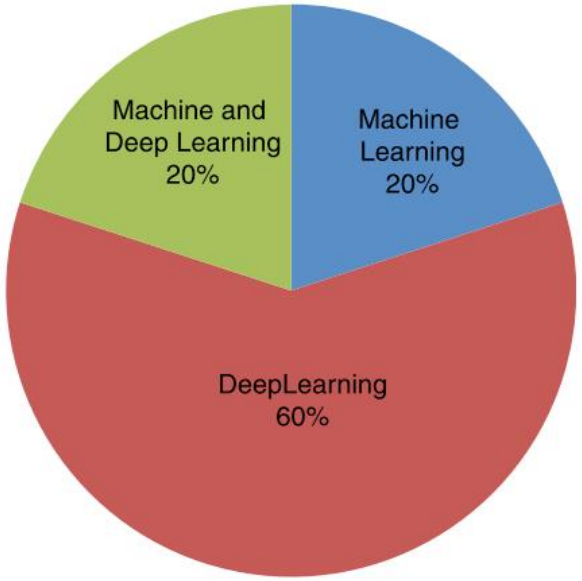
$$F1\text{-}score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Apruzzese, G., Andreolini, M., Marchetti, M., Colacino, V. G., & Russo, G. (2020). AppCon: Mitigating Evasion Attacks to ML Cyber Detectors. *Symmetry*, *12*

Xu hướng hiện nay

- Hiệu quả của NIDS dựa trên AI phụ thuộc nhiều vào quá trình huấn luyện với **tập dữ liệu phù hợp**
  - Với các model ML, thuật toán có thể được huấn luyện với các tập dữ liệu nhỏ để có kết quả tốt hơn. Nhưng với các tập dữ liệu lớn hơn, ML có thể không phù hợp trừ khi tập dữ liệu được gán nhãn
  - Vì việc gán nhãn tốn nhiều thời gian và chi phí, các giải pháp DL thích hợp hơn với các tập dữ liệu lớn. Những giải pháp này sẽ học và trích xuất thông tin hữu ích từ các tập dữ liệu thô
  - Tập dữ liệu lớn và đặc tính "sâu" của các thuật toán DL làm quá trình học tốn nhiều tài nguyên tính toán và thời gian
  - Model NIDS càng được huấn luyện nhiều thì nó sẽ phát hiện tấn công càng hiệu quả

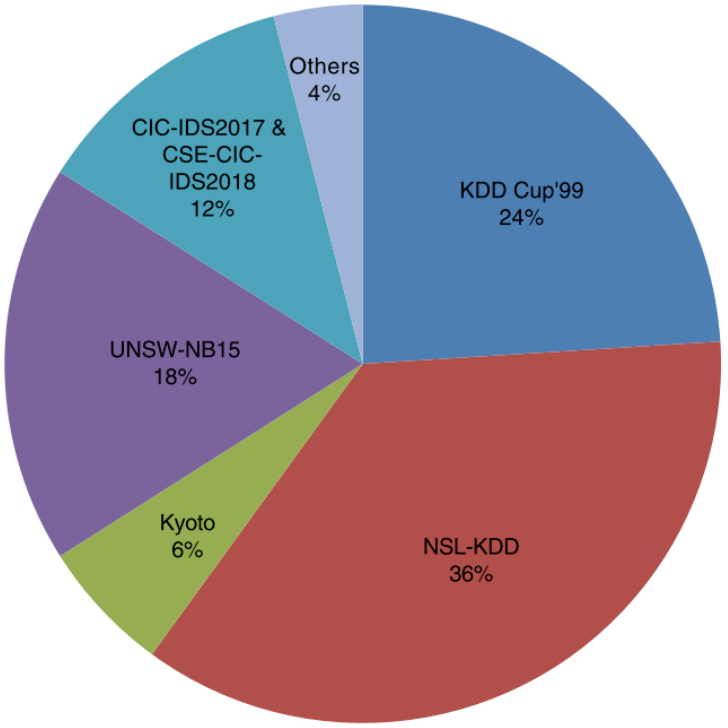
➔ **NIDS dựa trên DL** được ưa chuộng hơn so với các giải pháp ML



Phân bố các giải pháp IDS dựa trên ML và DL *(từ 2017 đến 2020)*

Xu hướng hiện nay (tt)

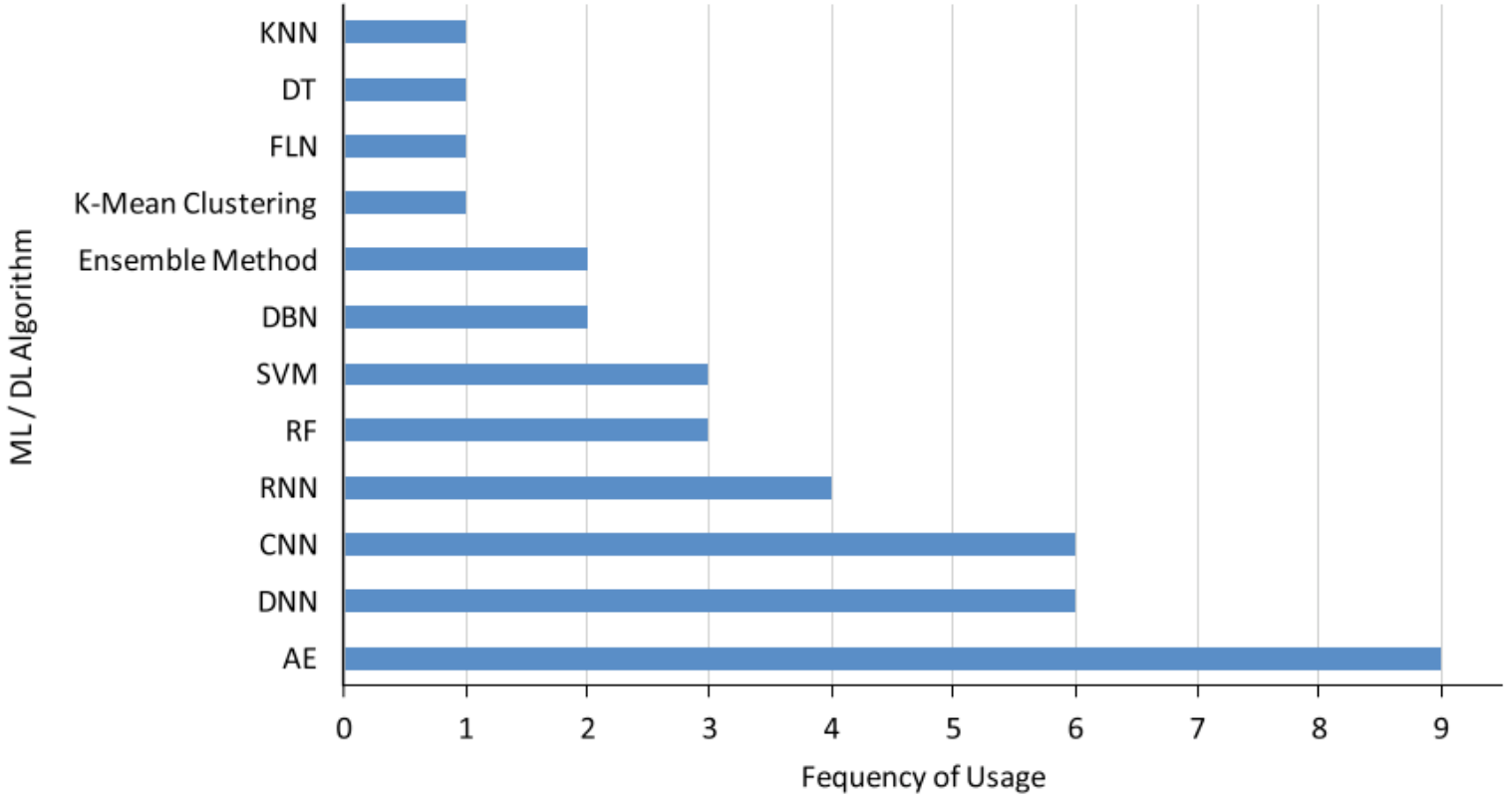
- Trong hầu hết các giải pháp, các model được kiểm tra trên các tập dữ liệu cũ hơn như KDD Cup'99 và NSL-KDD
  - Hiệu suất của model rất tốt trên các tập dữ liệu cũ có thể giảm khi áp dụng các tập dữ liệu mới hơn hoặc gần đây
- Thách thức:
  - **Mất cân bằng trong các lớp** ảnh hưởng đến tỉ lệ phát hiện và độ chính xác khi phát hiện các lớp tấn công có số lượng record ít
  - Model phức tạp có thể yêu cầu nhiều thời gian để huấn luyện → sự đánh đổi giữa độ phức tạp mô hình và cấu trúc "sâu" của DL



Phân bố các tập dữ liệu được sử dụng
(từ 2017 đến 2020)

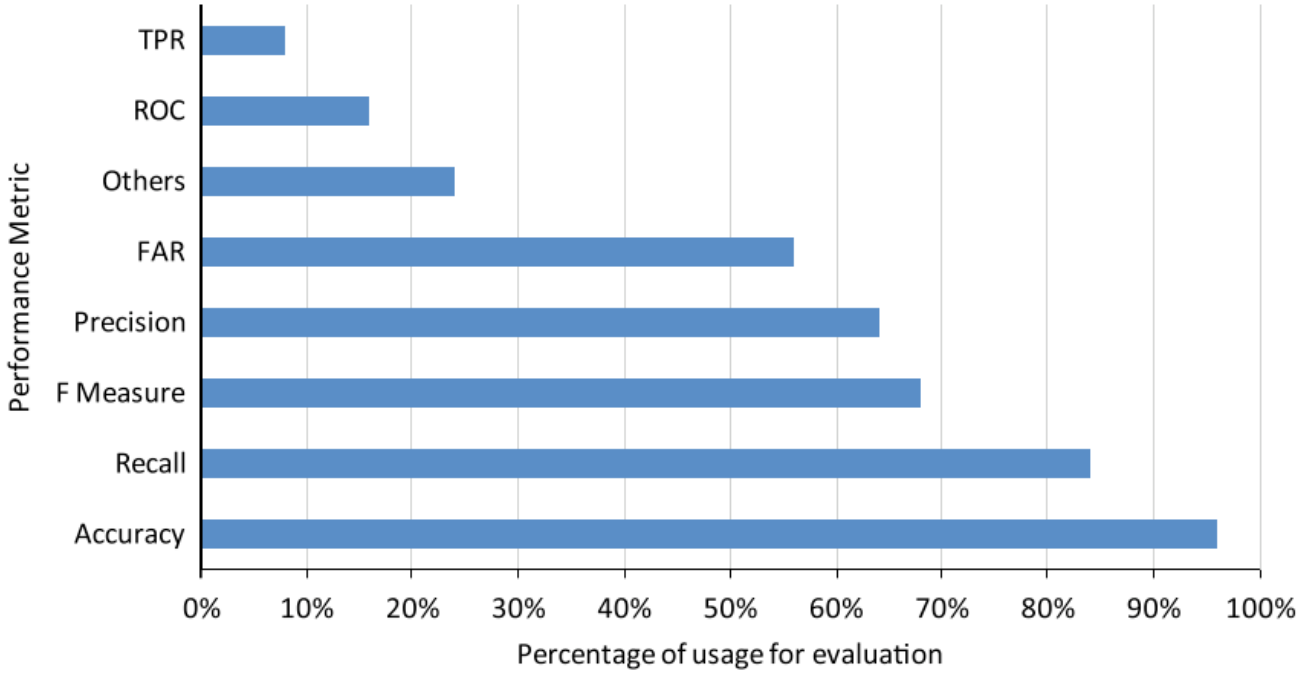
Xu hướng hiện nay (tt)

- Thống kê về các thuật toán machine learning và deep learning được sử dụng



Xu hướng hiện nay (tt)

- Thống kê về các Chỉ số đánh giá IDS được sử dụng



Các thách thức trong nghiên cứu

- Cần **tập dữ liệu cập nhật và có hệ thống**
- Độ chính xác trong phát hiện tấn công giảm do vấn đề **tập dữ liệu mất cân bằng giữa các lớp (class)**
  - **Giải pháp:** SMOTE, RandomOverSampler, adaptive synthetic sampling approach (ADASYN Algorithm),…
- Hiệu suất trong **môi trường thực tế**
  - Hầu hết các giải pháp được đề xuất đều được kiểm tra và kiểm chứng trong môi trường thí nghiệm với các tập dữ liệu public
- **Tiêu tốn tài nguyên** với các model phức tạp

  → cần thuật toán chọn lọc các thuộc tính hiệu quả
- Các IDS "nhẹ" cho ngữ cảnh IoT

Đọc thêm: Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J., & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. *Transactions on Emerging Telecommunications Technologies*, *32*(1), 1–29. https://doi.org/10.1002/ett.4150

Xu hướng trong tương lai

- Framework NIDS hiệu quả
  - Sử dụng tập dữ liệu cập nhật, có hệ thống và cân bằng
- Giải pháp cho các model phức tạp
- Sử dụng các thuật toán DL
  - Vẫn còn các thuật toán DL đáng quan tâm như Deep reinforcement learning, Hidden Markov Models, v.v...
- NIDS hiệu quả cho Cyber-Physical systems
  - Cyber-Physical Systems: **Supervisory Control and Data Acquisition (SCADA)** networks (smart grids, manufacturing industries) và mạng **Unmanned Aerial Vehicles (UAV)** – *Các mạng hỗ trợ phương tiện không người lái*

Hoạt động trên lớp

Tính các chỉ số đánh giá

Thực tế/dự đoán		Nhãn dự đoán	
		Tấn công (positive)	Bình thường (Negative)
Nhãn thực tế	Tấn công (positive)	55	45
	Bình thường (Negative)	50	850

- **Tính các chỉ số:**
  - Accuracy
  - FPR, FNR
  - Precision, Recall, F1-score

Chuẩn bị cho tuần sau…

- Hôm nay: **IDS dựa trên Machine Learning**
- Tuần sau: **Bảo mật cho các hệ thống dựa trên AI** *(Adversarial Machine Learning)*
  - Tài liệu:
    - Chio, C., & Freeman, D. (2018). *Machine Learning & Security* book (Chapter 8)
- Chuẩn bị nộp Báo cáo cuối kỳ **Đồ án môn học**

Câu hỏi/thắc mắc (nếu có)???

Today end, **See you next week!**