BÁO CÁO THỰC HÀNH

**Môn học: Hệ thống tìm kiếm, phát hiện và ngăn ngừa xâm nhập**

**Lab 05 – Học máy trong IDPS**

*GVHD: Đỗ Hoàng Hiển*

1. **THÔNG TIN CHUNG:**

*(Liệt kê tất cả các thành viên trong nhóm)*

Lớp: NT204.O21.ANTT.2

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ và tên** | **MSSV** | **Email** |
| 1 | Nguyễn Viết Dũng | 21520747 | 21520747@gm.uit.edu.vn |
| 2 | Lưu Thị Huỳnh Như | 21521242 | 21521112@gm.uit.edu.vn |
| 3 | Lê Đoàn Trà My | 21521149 | 21521149@gm.uit.edu.vn |

1. **NỘI DUNG THỰC HIỆN:[[1]](#footnote-2)**

|  |  |  |
| --- | --- | --- |
| **STT** | **Công việc** | **Kết quả tự đánh giá** |
| 1 | Yêu cầu 1.1 | 100% |
| 2 | Yêu cầu 2.1 | 100% |
| 3 | Yêu cầu 2.2 | 100% |
| 4 | Yêu cầu 3.1 | 100% |

**Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.**

BÁO CÁO CHI TIẾT

# **1. Yêu cầu 1.1:** Sinh viên tìm hiểu về tập dữ liệu KDD Cup 1999 và điền các kết quả tìm hiểu được vào form bên dưới.

|  |
| --- |
| **TÌM HIỂU VỀ TẬP DỮ LIỆU KDD CUP 1999**  Dữ liệu trong bộ dữ liệu KDD Cup 1999 là lựu lượng mạng đã được thu thập, phân tích, xử lý để lấy các thuộc tính và từ đó gán nhãn tương ứng với loại tấn công hoặc dữ liệu bình thường. Sinh viên tìm hiểu các phần sau:  1. Số nhóm tấn công: **4**  Kể tên các nhóm tấn công: **Dos (Denial of Service), Probe, R2L (Remote to Local), U2R (User to Root).**  2. Số kiểu tấn công: **22**  Kể tên các kiểu tấn công được gán nhãn: **back, buffer\_overflow, ftp\_write, guess\_passwd, imap, ipsweep, land, loadmodule, multihop, neptune, nmap, perl, phf, pod, portsweep, rootkit, satan, smurf, spy, teardrop, warezclient, warezmaster**.  3. Mỗi instance trong tập dữ liệu KDD Cup 1999 bao gồm **41** thuộc tính, cụ thể gồm các thuộc tính: **duration, protocol\_type, service, flag, src\_bytes, dst\_bytes, land, wrong\_fragment, urgent, hot, num\_failed\_logins, logged\_in, num\_compromised, root\_shell, su\_attempted, num\_root, num\_file\_creations, num\_shells, num\_access\_files, num\_outbound\_cmds, is\_host\_login, is\_guest\_login, count, srv\_count, serror\_rate, srv\_serror\_rate, rerror\_rate, srv\_rerror\_rate, same\_srv\_rate, diff\_srv\_rate, srv\_diff\_host\_rate, dst\_host\_count, dst\_host\_srv\_count, dst\_host\_same\_srv\_rate, dst\_host\_diff\_srv\_rate, dst\_host\_same\_src\_port\_rate, dst\_host\_srv\_diff\_host\_rate, dst\_host\_serror\_rate, dst\_host\_srv\_serror\_rate, dst\_host\_rerror\_rate, dst\_host\_srv\_rerror\_rate** |

# **2. Yêu cầu 2.1:** Sinh viên cài đặt WEKA, tìm hiểu và load một tập dữ liệu có định dạng .arff đơn giản có sẵn của WEKA.

- Mở file iris.arff có sẵn trong mục data của WEKA:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

→ Có thể thấy, tab Preprocess hiển thị các thông tin như Current relation, Attributes, Selected attribute, Class và Status.

+ ***Trường Current relation:*** Thông tin của tập dữ liệu hiện tại:



* Relation: tên của tập dữ liệu
* Instances: tổng số lượng mẫu trong tập dữ liệu
* Attributes: các thuộc tính của tập dữ liệu
* Sum of weights: tổng trọng số

+ ***Trường Attributes:*** Dùng để hiển thị các thuộc tính trong dataset

Ví dụ: Từ các hình trên, có thể thấy file iris.arff thì có 5 thuộc tính là sepallength (độ dài của đài hoa), sepalwidth (độ rộng của đài hoa) , petallength (độ dài cánh hoa), petalwidth (độ rộng cánh hoa) và class.

A screenshot of a computer

Description automatically generated

**+ *Trường Selected attribute*:** hiển thị tên của thuộc tính hiện tại và các thông số thống kê của mẫu sở hữu thuộc tính này như Minimum (giá trị nhỏ nhất), Maximum (giá trị nhỏ nhất), Mean (giá trị trung bình), StdDev (độ lệch chuẩn),…

A screenshot of a computer

Description automatically generated

**+ *Trường Class:*** Với mỗi attribute sẽ có tab vitualize dùng để hiển thị biểu đồ thống kê tương ứng với trường Attributes và Select attribute, số lượng các các mẫu có chung giá trị thuộc tính và đánh màu cho các mẫu có chung class.

A graph with red and blue squares

Description automatically generated

**+ *Trường Status*:** Hiển thị tình trạng của file. Chủ yếu có 2 trạng thái là đang xử lý (load) hoặc OK.



# **3. Yêu cầu 2.2:** Sinh viên lựa chọn 01 bộ phân lớp (classifier) bất kỳ và thực hiện khai thác trên tập dữ liệu đã chọn ở trên. Trình bày và giải thích kết quả.

- Trong cửa sổ WEKA Explorer, tại tab Classify lựa chọn bộ phân lớp **OneR -B 6**.

***\* Một số thông tin về bộ phân lớp OneR (One Rule):***

OneR (One Rule) là bộ phân lớp đơn giản dựa trên tầng xuất của mỗi thuộc tính để tạo ra nhiều rule phân loại khác nhau tương ứng với từng thuộc tính, kết quả sẽ là rule có sai lệch thấp nhất.

Ý nghĩa các flag:

-B: số lượng object tối thiểu trong 1 bucket

- **Test options**: Các tùy chọn được WEKA hỗ trợ để định nghĩa 2 tập dữ liệu huấn luyện và kiểm tra:

A screenshot of a computer

Description automatically generated

**+ Use training set:** Sử dụng tất cả dữ liệu đã load để tiến hành train và test. Điều này giúp kiểm tra hiệu suất và thời gian chạy của thuật toán.

**+ Supplied test set:** Cung cấp một tập dữ liệu kiểm tra riêng biệt, giúp đánh giá mô hình trên dữ liệu chưa từng được thấy trong quá trình huấn luyện.

**+ Cross-validation:** Folds là số dataset con được tạo ra; với folds = x, dataset sẽ được chia thành x phần, WEKA sẽ train trên x-1 dataset con và thực hiện test trên tập còn lại; thực hiện công đoạn này cho đến khi mỗi tập dataset con đều được lấy làm test set, điểm trung bình được ghi lại là thước đo của hiệu suất mô hình.

**+ Percentage split:** Tùy chọn này chia dữ liệu thành hai tập: tập train và tập test với tỉ lệ x, x là % được lấy làm tập train.

- Nhấn Start để chạy bộ phân lớp và kết quả:

A screenshot of a computer

Description automatically generated

**→** Classifer OneR được áp dụng cho dataset iris với 150 thực thể và 5 thuộc tính, test opion là cross-validation Flod 10.

**→** Mô hình được huấn luyện dựa trên thuộc tính “petalwidth”. Mô hình đã dự đoán đúng 144 trên 150 thực thể.

A screenshot of a computer

Description automatically generated

**→** Tổng quan về kết quả:

+ Correctly Classified Instance (Số lượng mẫu dán nhãn đúng) là 138 ứng với tỷ lệ 92%.

+ Incorrectly Classified Instances (Số lượng mẫu dán nhãn sai) là 12 ứng với tỷ lệ là 8%.

+ Giá trị Kappa statistic (đo lường mức độ thống nhất giữa dự đoán của mô hình và kết quả thực tế, dao động từ -1 đến 1): 0.88, cho thấy mô hình có độ chính xác cao.

+ Mean absolute error (MAE): sai số tuyệt đối trung bình là 0.0533.

+ Root mean squared error (RMSE): sai số bình phương trung bình là 0.2309.

+ Relative absolute error (RAE): sai số tuyệt đối tương đối là 12%.

+ Root relative squared error (RRSE): sai số bình phương tương đối là 48.9898%.

(Các chỉ số MAE, RMSE, RAE RRSE đo độ lệch giữa dự đoán của model và giá trị thật (giá trị càng nhỏ thì dự đoán càng đúng).

+ Total Number of Instances: Tổng số thực thể là 150.

→ Detailed Accuracy By Class là độ chính xác về hiệu suất theo từng lớp của mô hình.

Hàng cuối cùng là “Weighted Avg” : các chỉ số trung bình của các trọng số cho tất cả các lớp, với trọng số dựa trên số lượng mẫu của mỗi lớp.

→ Confusion matrix: đánh giá hiệu suất của mô hình phân loại.

Trong ma trận này:

+ Các hàng đại diện cho các lớp thực tế.

+ Các cột đại diện cho các lớp dự đoán bởi mô hình.

# **4. Yêu cầu 3.1:** Sinh viên lựa chọn 01 bộ phân lớp bất kỳ và thực hiện khai thác trên tập dữ liệu KDD Cup 1999. Giải thích và đánh kết quả.

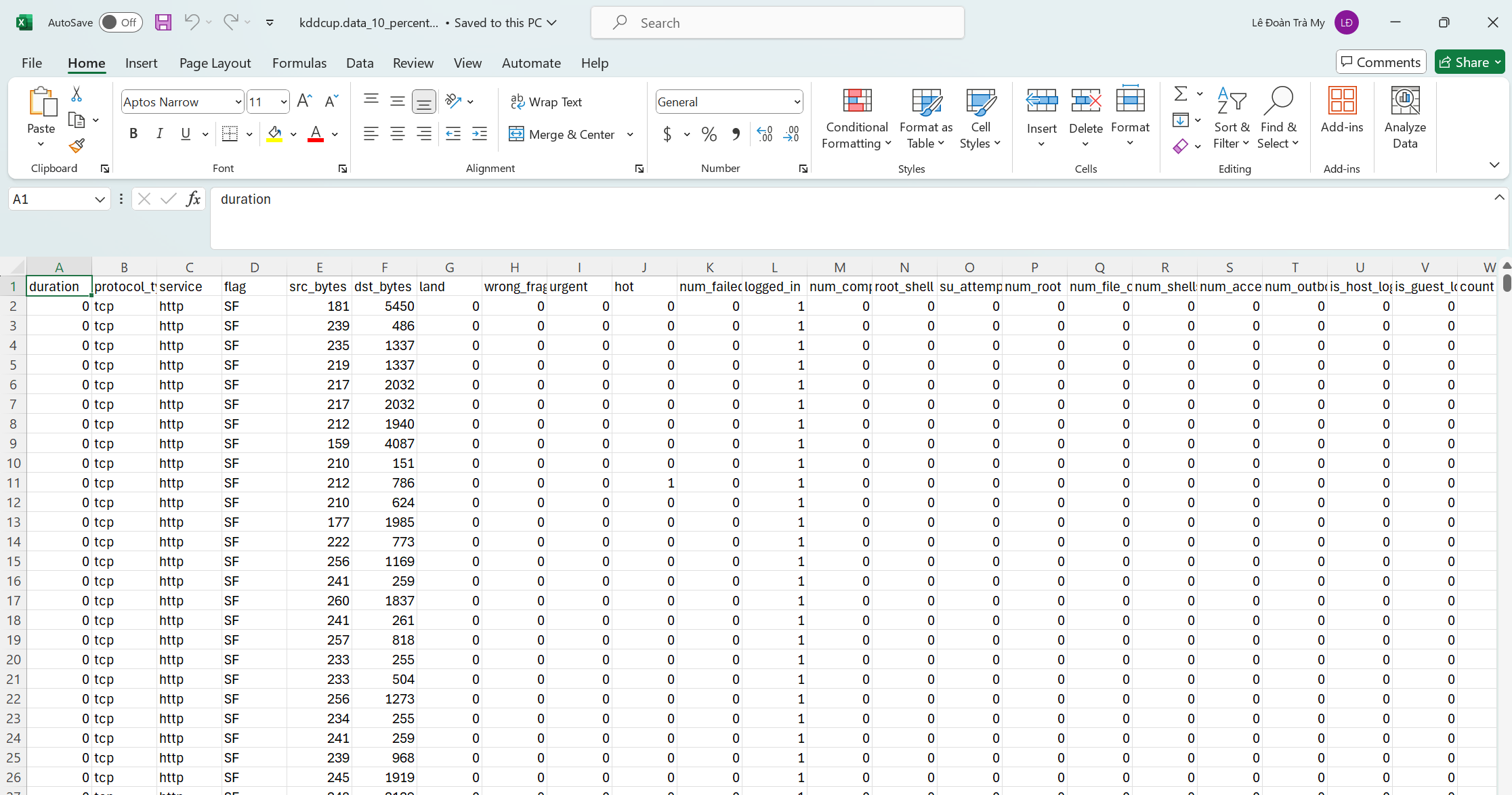
- Chuẩn bị dataset:

+ Tải các file cần thiết: kddcup.data\_10\_percent.zip và kddcup.names. Lấy danh sách các thuộc tính trong file kddcup.names và chuyển chúng thành một dòng lần lượt phân cách bằng dấu phẩy (,) rồi chèn vào đầu file kddcup.data\_10\_percent.

A screenshot of a computer

Description automatically generated

- Sử dụng Excel để mở tập dữ liệu, thao tác và lưu tại dưới định dạng file .csv:



- Tiến hành Load file vào WEKA

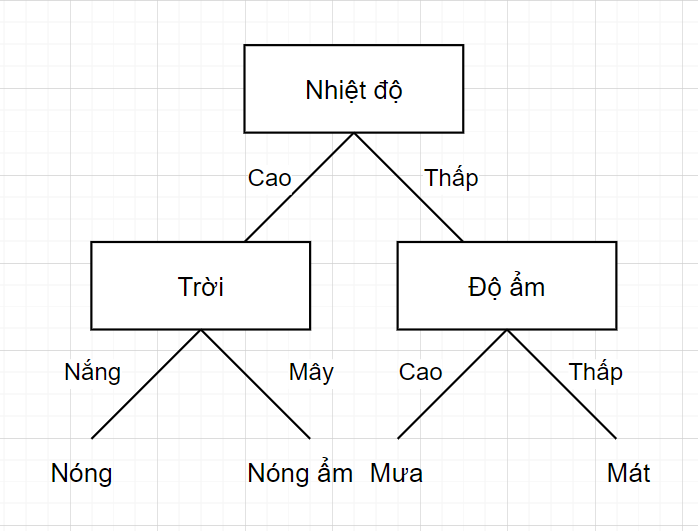
A screenshot of a computer

Description automatically generated

- Lựa chọn bộ phân lớp để thực hiện khai thác dữ liệu: Ở đây chọn bộ phân lớp J48.

***\* Một số thông tin về Bộ phân lớp J48:***

* Là bộ phân lớp được xây dựng dựa trên bộ phân lớp ID3 (Iterative Dichotomiser 3), sinh ra cây quyết định (Decision tree) dựa trên nguyên lý hoạt động của Lý thuyết thông tin (Information theory).
* J48 là một thuật toán phân loại dữ liệu được sử dụng để dự đoán lớp của các mẫu dữ liệu mới dựa trên các thuộc tính của chúng. Thuật toán này hoạt động bằng cách xây dựng một cây quyết định, là một cấu trúc dạng cây giúp phân chia dữ liệu thành các nhóm nhỏ hơn dựa trên các giá trị của các thuộc tính.
* J48 và Decision tree của nó:
  + Ví dụ như dự đoán thời tiết, J48 xây dựng 1 cây quyết định để phân loại dữ liệu. Cây này gồm các nút, nhánh con và lá cây. Mỗi nút đã diện cho một thuộc tính, nhánh con đại diện cho các giái trị có thể của thuộc tính đó và lá cây đại diện cho các lớp phân loại. Bắt đầu từ nút gốc của cây, thuật toán sẽ hỏi một câu hỏi về thuộc tính tương ứng với nút đó. Dựa vào câu trả lời, thuật toán sẽ di chuyển đến nhánh con phù hợp. Quá trình này được lặp lại cho đến khi đến một lá cây, đại diện cho lớp phân loại của mẫu dữ liệu.



* Tree Pruning: Sau khi xây dựng cây quyết định, J48 có thể thực hiện bước tỉa cây (pruning) để loại bỏ các nhánh không cần thiết nhằm tránh hiện tượng quá khớp (overfitting). Khái niệm pre-pruning là khi thuật toán kết thúc trước khi cây quyết định trở nên quá lớn hoặc trước khi nó trở thành cây trưởng thành. Điều này ngăn nguy cơ overfitting và giữ cho cây quyết định không trở nên quá phức tạp. Có ba điều kiện kết thúc nên được áp dụng khi xây dựng cây quyết định, thứ nhất là dừng thuật toán nếu cỡ mẫu quá nhỏ, thứ hai là giới hạn độ sâu của cây để ngăn nó trở nên quá lớn và thứ ba là không chia nút thành các nhánh nếu lỗi phân loại không giảm đủ.
* J48 là một thuật toán với các bước chính bao gồm chọn thuộc tính phân chia tốt nhất, tạo nút và phân nhánh đệ quy, xử lý các giá trị thiếu và tỉa cây để tránh quá khớp. Nó mạnh mẽ trong việc xử lý dữ liệu và có thể tạo ra các mô hình dự đoán hiệu quả.

- Ý nghĩa các flag:

+ -C: Ngưỡng tỉa cây: khi đạt ngưỡng nhất định sẽ bỏ đi nhánh trên decision tree.

+ -M: số mẫu tối thiểu trong mỗi node lá trong cây điều kiện

- Trong Test option, lựa chọn cross-validation vì dataset mất cân bằng.

- Nhấn Start và thu được kết quả:

A computer screen shot of a computer

Description automatically generated

→ Dataset có tổng cộng 494020 mẫu, 42 thuộc tính.

A screenshot of a computer screen

Description automatically generated

→ Model có tỷ lệ dự đoán chính xác rất cao:

+ Số lượng mẫu dán nhãn đúng: 493828 ứng với tỷ lệ 99.9611%.

+ Số lượng mẫu dán nhãn sai: 192 ứng với tỷ lệ là 0.0389%.

+ Giá trị Kappa statistic rất cao 0.9993 (xấp xỉ 1) cho thấy mô hình có độ chính xác rất cao.

+ Mean absolute error (MAE), Root mean squared error (RMSE), Relative absolute error (RAE), Root relative squared error (RRSE) rất nhỏ, lân lượt tương ứng 0, 0.0056, 0.0908%, 3.509%. Cho thấy độ chính xác dự đoán rất cao.

+ Kết quả có dấu “?” do một số class không có mẫu nào được đánh đúng dẫn tới trường hợp chia cho 0. Việc này xảy ra do bị mất cân bằng mẫu của mỗi lớp.

- Confusion matrix:

A group of small squares

Description automatically generated with medium confidence

**---HẾT---**

1. Ghi nội dung công việc, yêu cầu trong bài Thực hành [↑](#footnote-ref-2)