6

Lab

Học sâu trong IDS

Thực hành

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

Lưu hành nội bộ

A. TỔNG QUAN

A.1 Mục tiêu

- Hiểu được cách huấn luyện một IDS bằng học sâu.
- Nắm được cách sử dụng thư viện Tensorflow để xây dựng các mô hình học sâu.

A.2 Cài đặt môi trường

- Môi trường thực hiện: Google Colab[1].

[1] https://colab.google/

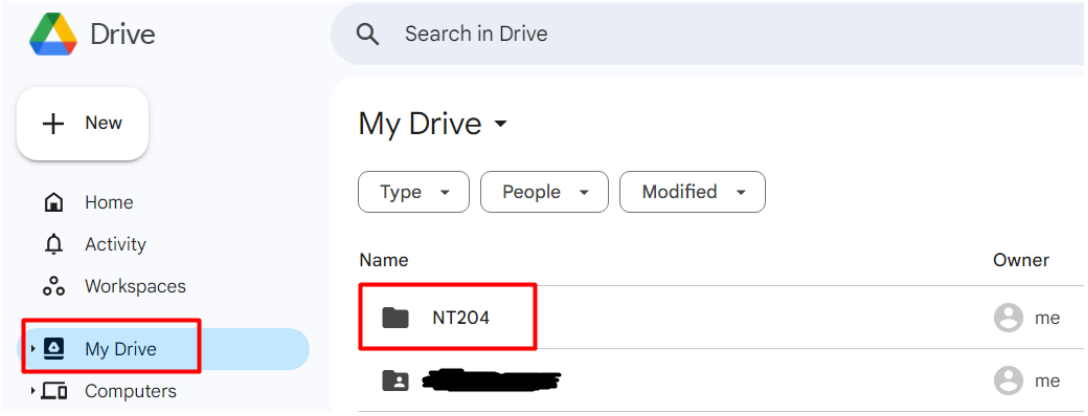
B. THỰC HÀNH

B.1 Sử dụng Google Colab

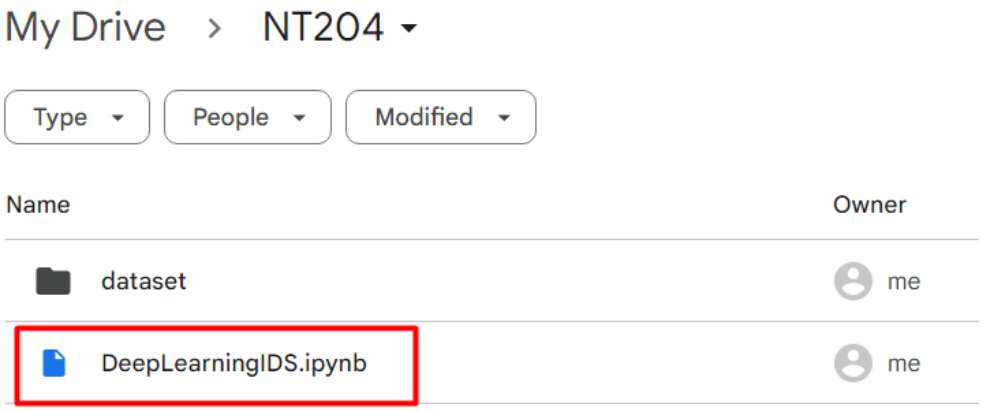
Sinh viên thực hiện các bước sau:

B.1.1 Chuẩn bị tập dữ liệu

– Bước 1: Trong mục **My Drive**, tạo thư mục **NT204**.



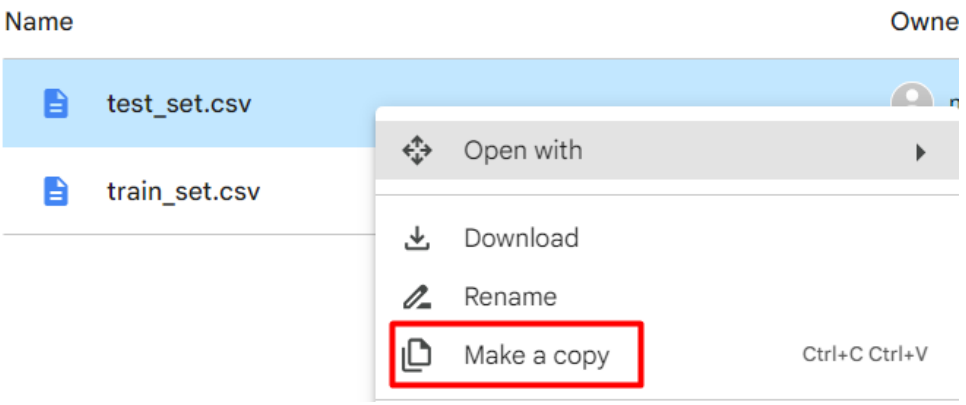
– Bước 2: Upload file **DeepLearningIDS.ipynb** (đã được cung cấp) lên thư mục **NT204**. Đồng thời, tạo một thư mục con có tên **dataset**.



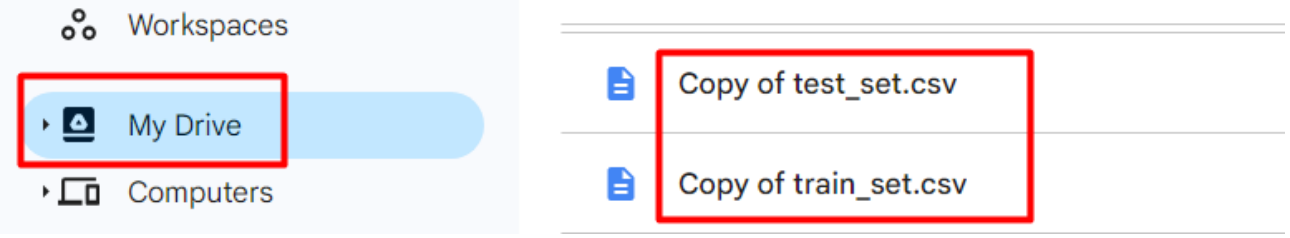
– Bước 3: Upload các tập train và test vào thư mục **NT204 > dataset**. Tập train đặt tên: train_set.csv, tập test có tên: test_set.csv.
Tải các tập dữ liệu tại:
https://drive.google.com/drive/folders/1nYyDqbmichotvI9eSOrKo-yRwtLUy_iM?usp=sharing.

Các bước thực hiện

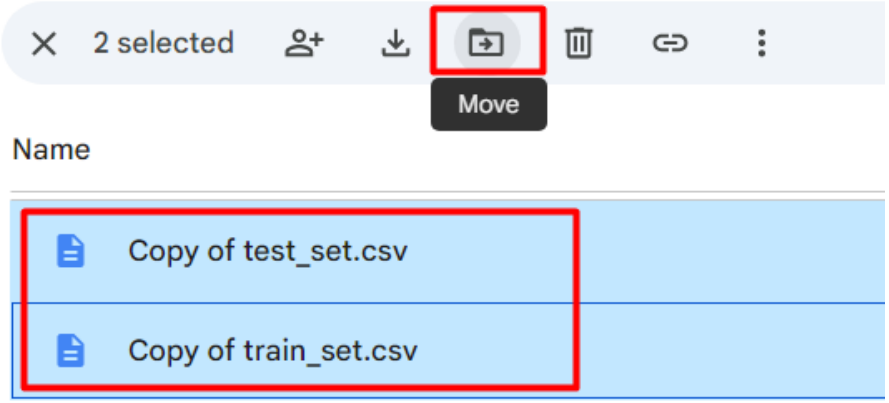
+ Để sao chép nhanh chóng các bộ dữ liệu bằng cách vào link được cung cấp phía trên, click chuột phải và từng file và chọn **Make a copy** (Tạo bản sao).



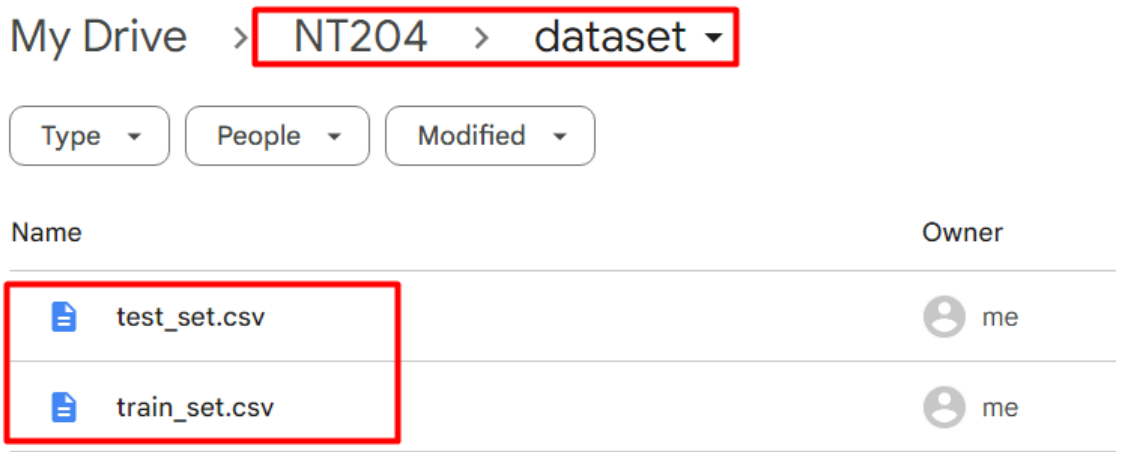
+ Các bản sao sẽ được tạo trong thư mục **My Drive** (Drive của tôi).



+ Vào My Drive và di chuyển (move) các tập dữ liệu và thư mục ***My Drive > NT204 > dataset***.



+ Đổi tên 2 file thành **test_set.csv** và **train_set.csv**.

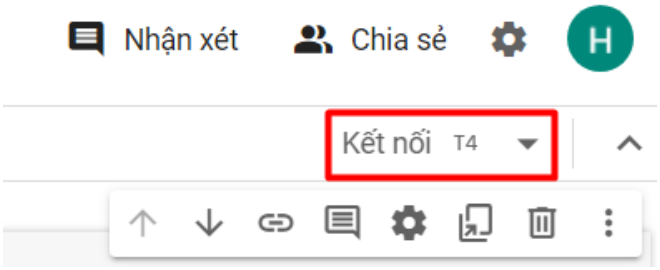


B.1.2 Mở mã nguồn bằng Google Colab

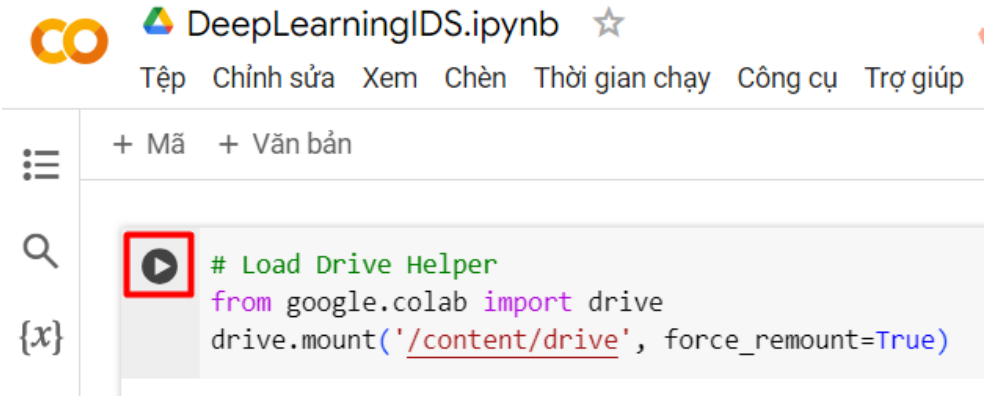
– Bước 1: Mở file **DeepLearningIDS.ipynb** trong thư mục **My Drive > NT204**.

Trong file đã có sẵn source code, sinh viên đọc các comment để hiểu thêm.

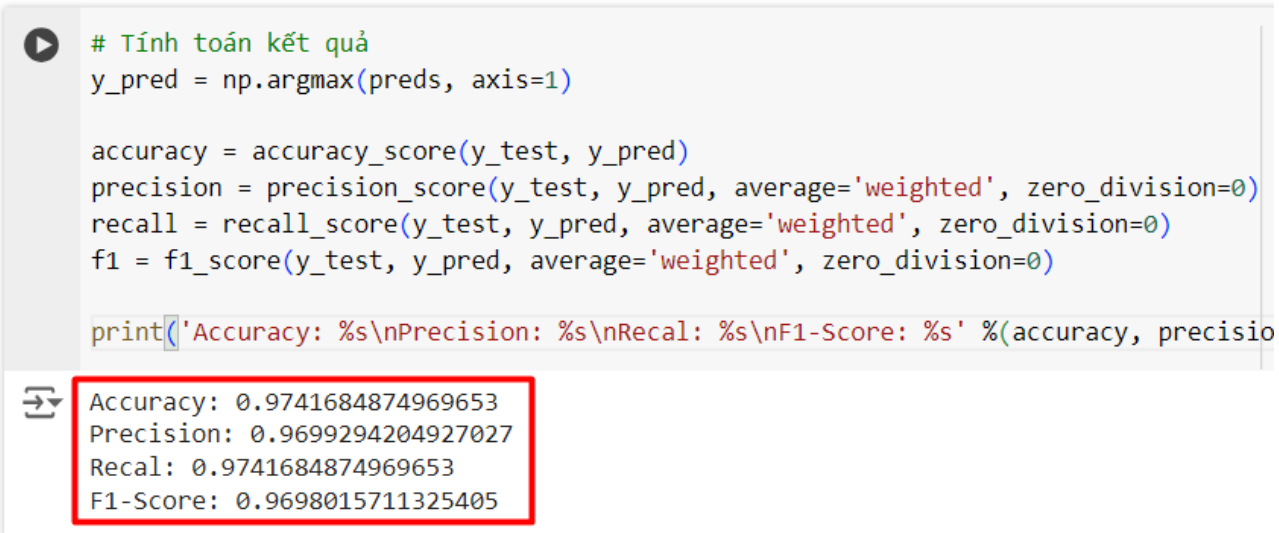
– Bước 2: Ở góc phía trên bên phải, click nút **Connect** (Kết nối) để tạo môi trường chạy code.



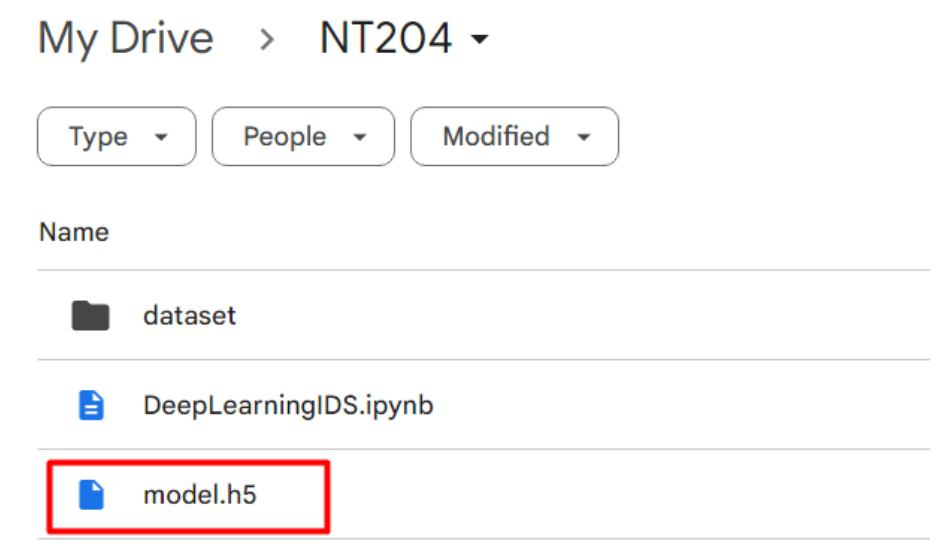
– Bước 3: Bấm nút **Run Cell** để chạy lần lượt từng khối code.



Sau khi chạy khối code cuối cùng, ta được kết quả như bên dưới:



Mô hình (file .h5) được lưu trong thư mục My Drive > NT204.



Xem thông tin của mô hình.

```
# Xem thông tin của mô hình
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 32)                1376

 dropout (Dropout)           (None, 32)                0

 dense_1 (Dense)             (None, 11)                363

=================================================================
Total params: 1739 (6.79 KB)
Trainable params: 1739 (6.79 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

B.2 Chỉnh sửa mô hình học sâu

Sinh viên tìm hiểu thêm về framework Tensorflow[2] và các kiến thức học sâu có liên qua để chỉnh sửa mô hình học sâu

> **Yêu cầu** Sinh viên sử dụng mô hình CNN hoặc RNN và các tham số liên quan để tăng độ chính xác (accuracy) của mô hình.
> Các kết quả (Accuracy, Precision, Recal, F1-Score) trên tập test **tối thiểu là 0.9770**.

Chỉnh sửa mô hình tại đây.

[2] https://www.tensorflow.org/tutorials/quickstart/beginner

```
# Tạo mô hình deep learning
model = tf.keras.Sequential(
  [
    tf.keras.layers.Dense(32, activation="relu", input_shape=(NUM_FEATURES,)),
    tf.keras.layers.Dropout(0.15),
    tf.keras.layers.Dense(NUM_CLASSES, activation='softmax')
])

# Compile mô hình
model.compile(optimizer='adam', loss = tf.keras.losses.CategoricalCrossentropy(), metrics=['accuracy'])
```

Chỉnh sửa batch size và số epoch tại đây.

```
# Huấn luyện mô hình
model.fit(x=x_train, y=y_train, batch_size=1024, epochs=10, shuffle=True)
```

Việc chỉnh sửa code cần lưu ý các vấn đề sau:

- Các bộ dữ liệu đã được tiền xử lý với Minmax Scaler.
- Không chỉnh sửa các file dữ liệu, chỉ được phép xử lý dữ liệu bằng code.
- Có thể chỉnh sửa batch size và epoch khi huấn luyện mô hình, nhưng không vượt quá **50 epoch**.
- Có thể chỉnh sửa hàm loss và optimizer.
- Tính toán trên đa nhãn (multiclass), không chuyển về 2 nhãn (binary class).
- Viết comment giải thích những khối code đã thêm hoặc chỉnh sửa.
- Tham số average của các hàm precision_score, recall_score, f1_score là **weighted** (giống code mẫu).
- Tham số huấn luyện (Trainable params) không vượt quá **500.000**. và không được sử dụng quá **3 Dense** layer.

C. YÊU CẦU

– Sinh viên tìm hiểu và thực hành theo hướng dẫn, thực hiện **theo nhóm**.

– Đảm bảo code phải chạy được trên Google Colab khi muốn thực nghiệm lại.

Hình thực báo cáo

– Sinh viên nén file mô hình (**model.h5**) và file **DeepLearningIDS.ipynb** (không xoá kết quả đã chạy) vào file ZIP với định dạng **[Mã lớp]-LabX_NhomY.ZIP**.

– Nộp file ZIP theo thời gian đã thống nhất tại courses.uit.edu.vn.

~HẾT~