**Project: Toxic comment classification**

**Task**

You are provided with a large number of comments which have been labeled by human raters for toxic behavior. The types of toxicity are:

* toxic
* severe\_toxic
* obscene
* threat
* insult
* identity\_hate

For the classification task, you need to predict a binary label (0 or 1) for each of the six possible types of comment toxicity (toxic, severe\_toxic, obscene, threat, insult, identity\_hate) for every id in the test set.

**File descriptions**

* **train.csv** - the training set, contains comments with their binary labels
* **test.csv** - the test set, you must predict the toxicity for these comments.
* **sample\_submission.csv** - a sample submission file in the correct format

**Submission File**  
The submission file must include a header and follow the format below, with the columns in the specified order:

id,toxic,severe\_toxic,obscene,threat,insult,identity\_hate

00001cee341fdb12,1,0,0,1,0,0

0000247867823ef7,1,0,1,0,1,1

00013b17ad220c46,0,0,1,0,0,0

00017563c3f7919a,1,1,1,0,0,0

00017695ad8997eb,1,0,0,1,0,1

…

**Evaluation**

The evaluation metric will be the f1\_score with average='macro'. Here is a evaluation Code example:

from sklearn.metrics import f1\_score

import numpy as np

# Example data

# Ground truth labels (true values)

y\_true = [

[1, 0, 0, 1, 0, 0], # For id 00001cee341fdb12

[1, 0, 1, 0, 1, 1], # For id 0000247867823ef7

[0, 0, 1, 0, 0, 0], # For id 00013b17ad220c46

[1, 1, 1, 0, 0, 0], # For id 00017563c3f7919a

[1, 0, 0, 1, 0, 1] # For id 00017695ad8997eb

# ....

]

# Predicted labels

y\_pred = [

[1, 0, 0, 1, 0, 0], # Predicted for id 00001cee341fdb12

[1, 0, 1, 0, 1, 0], # Predicted for id 0000247867823ef7

[0, 0, 1, 0, 0, 0], # Predicted for id 00013b17ad220c46

[1, 1, 1, 0, 0, 0], # Predicted for id 00017563c3f7919a

[1, 0, 0, 1, 0, 1] # Predicted for id 00017695ad8997eb

# ....

]

# Calculate the Macro F1 Score

macro\_f1 = f1\_score(y\_true, y\_pred, **average='macro'**)

print(f"Macro F1 Score: {macro\_f1:.4f}")