Toxic Comment Classification

- All the team members should work together.
- It's your responsibility to coordinate with your team members.
- We will only provide the resources related to the project and it's your responsibility to learn and do the project.
- If you face any issues during model implementation, search for answers in the Google before reaching out to us.

Instructions

1. Assignment must be implemented in Python 3 only.
2. You are allowed to use libraries for data preprocessing (numpy, pandas, nltk etc) and for evaluation metrics, data visualization (matplotlib etc).
3. You will be evaluated not just on the overall performance of the model and also on the experimentation with hyper parameters, data preprocessing techniques etc.
4. The report file must be a well-documented Jupyter notebook, explaining the experiments you have performed, evaluation metrics and corresponding code. The code must run and be able to reproduce the accuracies, figures/graphs etc.

Task: Given a comment, classify it as toxic or not

1. Given input text file ("train.csv" and “test.csv”) containing comments, with each row having its corresponding labels (toxic, severe_toxic, obscene, threat, insult, identity_hate) attached to it.
2. Download the dataset from the link Toxic Comment Classification
3. This task also requires basic pre-processing of text (like removing stop words, stemming/lemmatizing, any other of your choice).
4. You are required to build classification model (Logistic Regression, SVM, Naive Bayes, MLP, Decision Trees, Random Forest etc).

Task-1: Data Analysis (July- to July-10)

1. Import necessary libraries
2. Load dataset
3. Preprocess data
    - Removing stop words
    - Stemming/lemmatizing
    - Store the preprocessed comment
4. Tokenization
    - Tokenize the preprocessed comments using word tokenizer
5. Split the preprocessed data into train, and test set
    - Train data size = 80%
    - Test data size = 20%

Task-2 (July-11 to July-15)

You are required to build the classifiers to train the baseline models

1. Logistic Regression
2. SVM
3. Decision Trees
4. Naive Bayes
5. Random Forest

Use the codebase for your reference:
https://towardsdatascience.com/multi-label-text-classification-with-scikit-learn-30714b7819c5

Note: The codebase is not limited, you can build the models by your own.

Task-3 (July-16 to July-21)

Improve the baseline models using the transformer models.

References:

1. https://medium.com/mlearning-ai/toxic-comments-classification-8d8a9a9b99e6
2. https://github.com/DnyaneshT/Toxic-Comment-Classification-using-BERT/blob/master/my-bert-model.ipynb

Task-4 (From July-22)

- Prepare the final presentation
- Prepare code base of the project (we suggest the github repo)
- Ready for the final presentation