Choose a job you love, and you will never have to work a day in your life.

(Confucius)

Lab 2
Supervised Learning

In this lab, we apply supervised approaches to a dataset of hate speech[1]. The dataset consists of Wikipedia comments, labeled with the classes 'toxic, severe_toxic, obscene, threat, insult, identity_hate' or none.

The aim of the Lab is to develop a binary classification procedure that classifies text into hate speech / not hate speech. The minimal goal of the lab is to achieve a high F1-Score on the dataset. Note that the dataset is highly skewed regarding the classes: 89% of the texts are not hate speech. Hence, a simple baseline that classifies all texts as not hate speech has an accuracy of 89% already.

There exist a lot of different ways for model selection/improvement in order to achieve a high F1-Score. Here are some useful links for that:

- General info on model selection: https://scikit-learn.org/stable/model_selection.html
- Grid search (to get the best model parameters): https://scikit-learn.org/stable/modules/grid_search.html
- Analysis of learning curves: https://scikit-learn.org/stable/modules/learning_curve.html

It's encouraged that you try out a deep learning library of your choice for this lab. The easiest library to get started is Keras (www.keras.io). Other popular libraries are PyTorch (www.pytorch.org/get-started/locally) or Tensorflow (www.tensorflow.org).

Installation of the deep learning libraries:

- PyTorch: https://pytorch.org/get-started/locally/
- Keras: https://keras.io/#installation
- Tensorflow: https://www.tensorflow.org/install

[1] https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/data

Getting started with a deep learning library:

- PyTorch: https://pytorch.org/tutorials/beginner/deep_learning_60min_blitz.html
- Keras: https://keras.io/getting-started/sequential-model-guide/
- Tensorflow: https://www.tensorflow.org/learn

Research has shown that word embeddings can have a significant impact on the performance of deep learning algorithms. If you want to use these embeddings, then the NLP-library spacy should give you a good introduction:

- https://spacy.io/
- https://spacy.io/usage/vectors-similarity

Possible applications of the classifier:

- Command line interface that takes a text file as an argument and outputs a hate speech score.
- Take tweets of politicians, calculate a hate speech score for each and compare the results.
- What are typical (non-)hate speech words?
- A tool that highlights/color codes/flags hate speech in a document on the sentence or word level.
- Whatever cool idea you might have :)

If you run into an error regarding the stopwords file from the nltk library, you'll have to download it first:

1. Activate your virtual environment
2. Start the python interactive mode by just typing the command 'python'
   Inside your python interactive mode:

   ```
   import nltk
   nltk.download('stopwords')
   ```

3. finished, the stopwords list should be available now