CIS4930: Introduction to Multimodal Machine Learning in Python
Individual Coding Assignment 02
Due Mar 13, 2023, 11:59 pm

Sentiment analysis is a natural language processing (NLP) technique used to determine whether the speaker's sentiment behind the text is positive or negative. In this coding assignment, you will be provided with a text dataset. You will extract a set of linguistic features from the dataset and build supervised machine learning models for sentiment classification.

Data Description: This dataset is collected from online discussions on a popular social media platform. Please download the training and testing dataset here. The datasets contain three columns:

- Index: The index of the discussion thread.
- Sentiment: The sentiment of the discussion thread. 0: Negative, 1: Positive.
- Text: The textual content of the discussion thread.

What you need to do:

- **Step 1: Exploratory Data Analysis**. This stage is the very initial stage of your data analysis. You may want to know the size and sentiment distribution of the dataset. You may also want to examine if there are any missing values. This initial data analysis stage helps you to have a better understanding of the dataset before you build your sentiment classification models.
- **Step 2: Text Preprocessing**. You need to prepare your training and testing dataset. Specifically for this problem, you need to preprocess the discussion texts, you may want to convert all words into lowercase and remove digital numbers and special characters. Please refer to our slides and class discussions for a full list of text preprocessing steps.
- **Step 3: Linguistic Feature Extraction**. You will extract linguistic features from the processed texts. You may consider a wide range of features we covered in the class, including bag-of-words, tf*idf, word2vec, etc. You may also consider other word-embedding semantic features such as Glove or BERT, but these are not required.
- **Step 4: Build your sentiment classification model**. Provide the extracted set of linguistic features from the training dataset to your classification model. Note that this is a binary classification problem. You may want to start with

classical machine learning algorithms such as Logistic Regression, SVM, Naive Bayes, and Random Forest. You may also consider neural-network-based classifiers, such as multilayer perceptron, but these are not required.

- **Step 5: Model evaluation**. Evaluate your model performance with the provided testing dataset. Recall the evaluation metrics we covered in the class and select appropriate metrics for this problem. Please compare the performance of different classifiers using the same linguistic feature and the performance of the same classifier using different linguistic features. Finally, discuss your experimental results and submit the assignment report.