Project Description

LLM text dection

We will use the data of a competition that challenges participants to develop a model that can accurately detect whether an essay was written by a (middle/high school) student or by an LLM. Download the data from the challenge, which will comprise student-written essays and (well, toy) LLM-generated ones. The test data of the competition are expected to comprise essays generated by a variety of LLMs and which are hard to distinguish from the ones generated by students. In this assignment, however, you are expected to create your own evaluation data.

A. Data augmentation

- Prompt an LLM to generate essays, so that you balance the data (use both prompts provided by the challenge).
- Build text classifiers on the augmented data, using cross validation with appropriate classification evaluation metrics to assess them, and suggest the best performing one.
- Compute two scores per generated text, one reflecting the **maximum** and the other the **average** similarity of that text with student essays.
- Study the correlation between the similarity scores and the prediction probability of your best classifier for the generated texts; compute the prediction probability per text, by training the selected classifier on all except from that text, which is used a test instance (a.k.a. the leave-one-out cross validation setting).
- Based on your study so far, decide which generated texts should be discarded in order to improve the benchmark and yield a more robust classifier.

B. Learning curves

- Keep a test set apart and split the train data to portions (10%, ..., 90%, 100%).
- Train your best performing algorithm on each portion.
- Assess each trained instance on the test (the same across portions) and on the training data.
- Visualise the two curves (train, test), based on an appropriate evaluation measure, diagnosing weak and strong points of your classifier (a.k.a. the learning curves).
- Add a regressor to the plot, to estimate how many more texts you should generate to reach the "best" performance.

C. Clustering-based augmentation

- Use K-Means, based on an approprate text representation and the (estimated) optimum K, to cluster the generated essays, and then the student essays.

- Compare the cluster balance (number of instances per cluster) between the two clusterings.
- Yield a title per cluster, reflecting the topic of the texts included.
- Study the similarities between the two clusterings, by finding clusters comprising similar texts.
- Generate more texts (as in A) in order to better balance your clusters.
- Re-train your best-performant classifier on the new data (or a careful selection of them) and analyze the benefits of using clustering to improve the classifier.