**Hands-on Assignment 07**

**Use CLIP models for zero-shot classification**

**Due Date: See web**

**Objective:**

In the CLIP tutorial, you learned how to use CLIP for zero-shot classification on the CIFAR100 dataset. For this assignment, you will apply the CLIP model to another dataset from [the list provided at CLIP data prompts](https://github.com/openai/CLIP/blob/main/data/prompts.md), excluding CIFAR100. You will utilize one of the CLIP models specified in the [CLIP model script](https://github.com/openai/CLIP/blob/dcba3cb2e2827b402d2701e7e1c7d9fed8a20ef1/clip/clip.py#L30) to perform zero-shot prediction on your chosen dataset.

**Assignment Details:**

1. Dataset and Model Selection:

* Choose a dataset from the list provided in the [CLIP data prompts URL](https://github.com/openai/CLIP/blob/main/data/prompts.md).
* Select one available CLIP model for your zero-shot prediction task.

1. Prompt Template Implementation:

Implement zero-shot prediction using two different sets of prompt templates:

* Simple Template: Use a straightforward template for each class, formatted as “a photo of a {CLASS}.”
* Ensemble Template: Utilize a combination of multiple prompt templates for each class as suggested in the CLIP data prompts documentation. There is more than one way for implementing the use of various prompt templates in the code for CLIP zero-shot prediction, and try to explore the most efficient one (though, the efficiency of your implementation will not influence your grade) .

1. Reporting:

Write a simple report for the submission. The report should include:

* Summary: Begin with a clear statement of the chosen dataset and CLIP model.
* Methodology: Describe the prompt templates used and detail your implementation strategy for employing multiple prompt templates per class.
* Results: Provide a comparative analysis of the accuracy achieved with each set of prompt templates. Highlight key findings and discuss any patterns observed.
* Visualization: Include visualizations of several example predictions, focusing particularly on instances where misclassification occurred. These visualizations can mimic the style seen in the CLIP tutorial.

The report must be self-contained, clear, and complete without supplementary code, as the evaluation focuses solely on the quality and completeness of the report.

**Notes**:

1. Similarity scores will be computed for this assignment and **similarity penalty will be applied** (details see the course webpage).

Directly copying a batch of Python outputs to the report may cause a high similarity score of your submission. In general, we **will not** accept requests for exemptions from the similarity penalty due to the duplication of Python outputs.

1. Submission: Submit a report in **docx or pdf** format via Canvas. The report should be named as **Student\_ID\_HA7\_CLIP.docx/pdf**.

There is **no need** to submit your code.

*Copyright Statement: The materials provided by the instructor in this course are for the use of the students enrolled in the course. Copyrighted course materials may not be further*  
 *disseminated.*