LING 574 HW9

Due 11:59PM on June 9, 2025

In this assignment, you will

- Develop understanding of prompting and in-context learning
- Implement some of the components of applying ICL to our sentiment analysis task
- Explore the impact of those choices on task performance, and compare to previous models

All files referenced herein may be found in `/mnt/dropbox/24-25/574/hw9/` on patas.

1 Prompting and in-context learning [30 pts]

Q1: In-context learning

- What is the primary difference between in-context learning (ICL) and fine-tuning? In particular, how are predictions made in the former? Please answer in 1-2 sentences. [5 pts]
- What is one unintuitive aspect of ICL that you have learned about in this course? Please answer in 1-2 sentences. [5 pts]

Q2: Post-training

- What are examples of two post-training methods? [2 pts]
- How do these methods differ from 'traditional' fine-tuning directly on a downstream task? [3 pts]
- Why has post-training been applied to base pre-trained models in recent years? What types of behaviors does it encourage? Please answer in 1-2 sentences. [5 pts]

Q3: Documentation of OLMo data For the coding portion of this assignment, we will be using two models from the OLMo (Open Language Model) project, namely `OLMo-2-0425-1B` and `OLMo-2-0425-1B-SFT`. You can find more information about these models at this release page, this earlier blog post, and this paper (clickable links). Based on these resources:

- Provide a brief description of the *pretraining data* used for the two models. How large is it, and what types of data does it include? Please answer in 1-2 sentences. [5 pts]
- Did you find it easier or harder to answer the previous question as compared to writing a data statement for SST in HW1? Why? [5 pts]

2 Implementing an ICL-based Sentiment Classifier [15 pts]

In the coding portion of this assignment, you will implement a model for sentiment analysis on the SST dataset, using a pretrained transformer decoder.

Q1: Implement a new template In `olmo.py`, you will find a basic template for one example of the SST task in (`template_basic`). Please implement a new template, of your choice, in `template_complex`. Please describe here what you chose to do, including one example. [5 pts]

Q2: Implement a new prompt In `olmo.py`, you will find a basic prompt for the SST task (`prompt_basic`). This simply concatenates the template as applied to the in-context examples and the new to-be-predicted review. Please implement a new prompt in `prompt_complex`. We recommend, minimally, pre-pending a task description and/or instruction to the basic prompt. Please describe here what you chose to do, including one example. [5 pts]

Q3: Implement a rating extractor In ICL, a model generates text in response to a prompt. In order to use this setup as a classifier, we need to extract a label from the generated text. Please implement `extract_rating` in `olmo.py`. This function should take as input the generated text and return a rating (1-5) for the sentiment of the review. You may develop any method you like, but we suggest as a default using a regular expression to extract the first number in the text. Please describe here what you chose to do, including one example. [5 pts]

3 Running the Classifier [25 pts]

`olmo.py` contains a basic loop for SST dev-set classification, by prompting a pre-trained transformer language model. NB: we are running on a smaller subset of SST (256 randomly-selected dev examples) for speed reasons, since we will ask you to do many variations.
`hw9.condor` is a job file that calls `run_patas.sh`. Please use this to run your homework, both to use Condor instead of the head-nodes directly, but also because it eliminates a few nodes that are not compatible with the course's new conda environment.
We are using two models from the OLMo (Open Language Model) project, namely `OLMo-2-0425-1B` (0425 refers to the release date, April 25, and 1B to the number of parameters) and `OLMo-2-0425-1B-SFT`. The latter starts from the former version, but then undergoes SFT (i.e. supervised fine-tuning).

Q1: Many runs. Please fill out the table below with the results of running `olmo.py` with the corresponding arguments. You can find the names for the command-line arguments in the `argparse` section of `olmo.py`. The first row has a command already filled in for you in `run_patas.sh`.
NB: this is 12 runs. Even with the reduced dev set, these can still take some time to run (especially with longer prompts and $k > 0$). Because of this, we strongly recommend starting early so that all of your runs finish with enough time before the deadline to answer the questions below. [5 pts]

Q2: Qualitative analysis

- What trends, if any, do you see in the accuracy table above (e.g. do any of the hyper-parameter choices seem to have a particularly large imapct)? What do you think is causing these trends? Please answer in 2-3 sentences. [7 pts]

Model	Template	Prompt	k	Accuracy
OLMo-2-0425-1B	basic	basic	0	
OLMo-2-0425-1B	basic	complex	0	
OLMo-2-0425-1B	complex	basic	0	
OLMo-2-0425-1B	complex	complex	0	
OLMo-2-0425-1B	complex	basic	4	
OLMo-2-0425-1B	complex	complex	4	
OLMo-2-0425-1B-SFT	basic	basic	0	
OLMo-2-0425-1B-SFT	basic	complex	0	
OLMo-2-0425-1B-SFT	complex	basic	0	
OLMo-2-0425-1B-SFT	complex	complex	0	
OLMo-2-0425-1B-SFT	complex	basic	4	
OLMo-2-0425-1B-SFT	complex	complex	4	

- For each run, the script also outputs the first batch of prompts, model generations, extracted labels, and true labels. Please include the first 10 examples from (a) the run with the highest accuracy and (b) the run with the lowest accuracy in your submission, and describe in 2-3 sentences what you see in these examples. Do you see any trends in what constitutes a good versus a bad prediction from the system? You may also browse and reference other runs' examples as well if you'd like. [8 pts]

Q3: Comparison to Fine-tuned Classifier In 2-3 sentences, please explain what differences you see in the classification decisions by this model using ICL and the small, fine-tuned, pretrained transformer that you trained in HW8. What do you think may be causing these effects (or lack thereof)? [5 pts]

Submission Instructions

In your submission, include the following:

- readme.(txt|pdf) that includes your answers to §1, §2, and §3.
- `hw9.tar.gz` containing:
  - run_hw9.sh. This should contain your run commands for §3 above.
  - olmo.py