CS 839: Foundation Models
Homework 1

Instructions: Read the two problems below. Type up your results and include your plots in LaTeX. Submit your answers in two weeks (i.e., Oct. 3 2024, end of day). You will need a machine for this assignment, but a laptop (even without GPU) should still work. You may also need an OpenAI account to use ChatGPT, but a free account should work.

1. NanoGPT Experiments. We will experiment with a few aspects of GPT training. While this normally requires significant resources, we will use a mini-implementation that can be made to run (for the character level) on any laptop. If you have a GPU on your machine (or access to one), even better, but no resources are strictly required.

- 1. *Clone Karpathy's nanoGPT repo* (https://github.com/karpathy/nanoGPT). We will use this repo for all the experiments in this problem. Read and get acquainted with the README.
- 2. *Setup and Reproduction.* Run the Shakespeare character-level GPT model. Start by running the prep code, then a basic run with the default settings. Note that you will use a different command line if you have a GPU versus a non-GPU. After completing training, produce samples. In your answer, include the first two lines you've generated.
- 3. *Hyperparameter Experimentation.* Modify the number of layers and heads, but do not take more than 10 minutes per run. What is the lowest loss you can obtain? What settings produce it on your machine?
- 4. *Evaluation Metrics.* Implement a *specific* and a *general* evaluation metric. You can pick any that you would like, but with the following goals: Your specific metric is meant to capture how *close* your generated data distribution is to the training distribution. Your general metric need not necessarily do this and should be applicable without comparing against the training dataset. Explain your choices and report your metrics on the settings above.
- 5. *Dataset.* Obtain your favorite text dataset. This might be collected data by a writer (but not Shakespeare!), text in a different language, or whatever you would prefer. Scrape and format this data. Train nanoGPT on your new data. Vary the amount of characters of your dataset. Draw a plot on number of training characters versus your metrics from the previous part. How much data do you need to produce a reasonable score according to your metrics?
- 6. *Fine-tuning.* Fine-tune the trained Shakespeare model on the dataset you built above. How much data and training do you need to go from Shakesperean output to something that resembles your dataset?

2. Prompting. We will attempt to see how ChatGPT can cope with challenging questions.

- 1. *Zero-shot vs. Few-shot.* Find an example of a prompt that ChatGPT cannot answer in a zero-shot manner, but can with a few-shot approach.
- 2. *Ensembling and Majority Vote.* Use a zero-shot question and vary the temperature parameter to obtain multiple samples. How many samples are required before majority vote recovers the correct answer?
- 3. *Rot13.* In this problem our goal is to use Rot13 encoding and 'teach' ChatGPT how to apply it. You can use rot13.com to quickly encode and decode. Also read about it at https://en.wikipedia.org/wiki/ROT13. Our goal is to ask questions like

  What is the capital of France?,

  but encoded with Rot13, i.e.,

  Jung vf gur pncvgny bs Senapr?,

- What do you obtain if you ask a question like this zero-shot? Note: you may need to decode back.
- What do you obtain with a few-shot variant?
- Provide the model with additional instructions. What can you obtain?
- Find a strategy to ultimately produce the correct answer to an encoded geographic (or other) question like this one.