Astra Research Exercises - Owain

Goal for research exercise

The goal is to investigate how well an LLM can articulate in natural language rules that it uses for a classification task.

Specifically, suppose T is a classification task where inputs are labeled based on a rule that is easy to express in natural language for humans. Are there tasks T that LLMs can learn very accurately (given sufficient examples) without being able to articulate the rule they have learned?

Terminology and examples

We use the term "classification task" following standard usage in ML. In this project, we focus on classification tasks where the inputs are strings.

For each classification task, there's a **classification rule** that determines how to classify each input. We define an **articulation** of a rule as a complete description of the rule in natural language.

Example 1

Here are some text inputs with classification labels:

```
Input: "the cat sat on the mat" Label: True
```

```
Input: "THE DOG RAN IN THE PARK" Label: False
```

```
Input: "THE mat sat on the cat" Label: False
```

```
Input: "the house is cold" Label: True
```

Here the classification rule can be articulated as:

"The input is labeled as 'True' iff the input is all lowercase".

This rule is simple to articulate in natural language. (Note that I only gave four labeled examples, which is not enough to learn the classification rule with high confidence.)

Example 2

Another classification task is to determine if a random paragraph is taken from the news section of Newspaper A or Newspaper B. This may be a challenging task if the two newspapers have similar style and content, such as the NYT and Washington Post. Still, I expect that LLMs could do reasonably well on this task with sufficient training data.

Note that it may be difficult or impossible to articulate a rule for this classification task using a short sentence in natural language. This is because the features that distinguish the classes are statistical in nature and relatively subtle. This makes classification tasks like Example 2 less good for this exercise, because the goal of articulating the classification rule may not be possible.

Research Exercise: What you need to do

You should start by focusing on in-context learning. That is, the LLM learns to classify examples in-context (few-shot learning) and learns to articulate in-context. If you have time, you are welcome to try finetuning but it is not necessary.

Which LLM should you use? My general recommendation is to use stronger LLMs. These include OpenAI's models via the OpenAI API, Claude via the Anthropic API, or the strongest open source models. In particular, I'd avoid using the smallest Llama-1 model.

Step 1. Find classification tasks that are learnable in-context

Find or create a set of classification tasks where your LLM performs well in-context.

Clarification:

"*In-context*" means using instructions and few-shot labeled examples of the classification task.

"*Performing well*" means getting >90% accuracy on held-out (in-distribution) examples.

Further points:

- Your tasks should have a classification rule that is simple to articulate for humans. For example, "the input is all lowercase" (see Example 1) or "the input contains a number". Feel free to come up with your own ideas for classification rules.
- You need to choose a space of inputs. These could be sentences, random strings, paragraphs, etc. They could be generated by an LLM or taken from an existing dataset or scrape.
- The goal is to test the LLM's ability to articulate classification rules. For that purpose, the more distinct classification tasks you have (where the LLM performs well) the better you can test articulation. Conversely, if you only have 3 classification rules, the LLM might do well at articulation by chance. So it's good to have many classification rules, without neglecting steps 2 and 3.

Step 2. Test the LLM's ability to articulate the rules

In Step 1, you found some set of rules S that your LLM learns to classify well from examples. Now the goal is to evaluate if the LLM can articulate these rules. (Note: be careful that your instruction in Step 1 did not give away information that makes the articulation in Step 2 trivial)

trivial).

You can test articulation either with multiple-choice (where the actual rule is one of a set of options) or with free-form generation. The free-form generation is harder. If you succeed at multiple-choice, focus on getting free-form generation to work.

Try different variations on your prompts to improve performance on articulation. For example, you can vary the instructions, the few-shot examples, and any use of chain-of-thought.

Step 3. Investigating faithfulness

Let's say your LLM successfully articulates a set of rules that it learns in context. (In other words, it succeeds at Step 1 and Step 2). Does mean the Step 2 faithfully explains the LLM's behavior in Step 1? What further tests could you do to investigate that? (Note that "faithfulness" is a term of art in AI safety. You may want to look at some related works if you are unfamiliar with it.)

Another possibility is that your LLM fails at Step 2 even though the rule is relatively simple. In this case, can you show evidence that the LLM can articulate or understand the rule in other contexts? If so, then the LLM's failure in Step 2 can be seen as a kind of "dishonesty". Try to understand what factors influence this dishonesty.

Some further guidance:

- It's really important to get to at least Step 2. If you are struggling with Step 1, you can either try classification rules or accept slightly less than 90% accuracy.
- You do not need to pursue these steps in linear order. Instead, you can think about them in parallel (especially steps 1 and 2).
- You are welcome to use finetuning, either for the classification or articulation tasks, but note that it may be challenging given the overall time constraints.
- You are welcome to use LLMs and any online materials to help. **However, do not work with anyone else on this project. This should be your own work.**

Output

Write a report on your findings. The report should include examples of the prompts, classification tasks, and model articulations. It should display quantitative results using figures or tables. The report should begin with an abstract that summarizes your methods and findings. Try to explain things as clearly as possible, and include complete axis labels and captions for your figures. The report should include a link to a Github repo with your code. The report should be in pdf format. You can use Latex but it's not required.

If you are uncertain or confused about any results (e.g. because they were slightly rushed), please say so in the report. Think of this as a report you are writing to your research collaborators (not yet for public consumption).

Timeframe

This exercise can be completed in two full days of work.