The University of Texas at Austin
School of Information

Week Four

Iteration and Annotation

Prompt Engineering

Mick McQuaid

mcq@utexas.edu

University of Texas at Austin

03 Feb 2025

quarto

Week Four

Agenda

- Presentations: Tianyi, Srishti, Puneeth
- News
- Review whatialreadyknow (Ishwari)
- Q&A on eB
- Q&A on m1
- Iteration
- Annotation
- Work time

Presentations

News

WhatIAlreadyKnow (Ishwari)

eB

Specification

- Assemble a bibliography in BibTeX format
- The subject is PDF remediation
- PDF remediation is the process of making a PDF accessible to people with disabilities
- Most of the literature is about research papers students are required to read
- I'm familiar with this literature, making it easier for me to detect hallucinations
- I can also judge the importance of the papers listed
- A recent bibliography in this area listed 39 sources, of which only about 20 were relevant
- Document your process!

m1

Deliverable

- A short qmd / html document describing the domain
- The doc should include specification of whether you plan to run locally or using a cloud service
- The doc should include a discussion of the possible datasets you might use (the actual dataset will be due in m2)
- You are not required to stick with the directions you give here, but this should be your current best guess of what you plan to do
- Examples: chatbot to emulate a foreign leader; chatbot to triage banking problems; chatbot to analyze tweets

Iteration

How do you iterate?

- Do you document what you do?
- Do you use any prompt engineering tool to redeploy queries?
- How many times do you iterate?
- How do you know when you're done?
- There are many such worthwhile questions!
- Let's play with a prompt engineering tool to learn more

Agenta

- https://github.com/Agenta-AI/agenta
- Let's pause to see a couple of Agenta videos
- ⟨ agenta intro ⟩
- ⟨ agenta quick start ⟩
- Let's try the workflow in the second video
- You can install it on your machine but I'll use the cloud version
- How many have Docker already installed?

Example workflow

- Visit https://github.com/Agenta-AI/agenta
- Select Agenta Cloud
- Set up an account
- I tried using `example_tweets_test` without success, so I switched to `example_tweets_test_tiny`
- Follow me through the workflow

Annotation

Example

- We'll work with *Potato*, a current experimental annotation tool
- It's described in a recent paper, Pei et al. (2023)
- It's an open source project
- It's written in Python
- It's available on GitHub
- paper at https://arxiv.org/abs/2212.08620

Getting started

- Clone the repository
- Install the dependencies
- Run the tool
- Sounds simple, right?
- Unfortunately, the instructions don't work
- I think they are out of date
- We'll do it our own way

Part one

```
mkdir labeling && cd labeling
git clone https://github.com/davidjurgens/potato.git
cd potato
pip install -r requirements.txt
python potato/flask_server.py start project-hub/politeness_rating/configs/politeness.yaml -p 80
```

Part two

- Visit http://localhost:8000
- You should see a login screen
- You have to create an account
- Click on the “Create an account” link
- Enter an account name and a password, it doesn’t really matter what

Part three

- Log in with your new account
- You should see an intro, then an instance to annotate
- There should be fifty instances to annotate
- Try to annotate them all
- You can just enter a number from 1 to 5, it doesn't matter what
- Try not to give them all the same number

Part four

- Do the demographics; again, they don't matter, just enter something
- Click on the "Submit" button
- Now you want to see your results
- They are in a folder called `potato/project-hub/politeness_rating/annotation_output/full/<your account name>/`
- Kill the server with `Ctrl-C` and use that window to navigate to the folder

Part five

- There are several files in the folder
- `annotation_order.txt` is a list of the instances you annotated in order
- `assigned_user_data.json` is a list of the instances you annotated, with the annotations you made
- `annotated_instances.jsonl` is a list of the instances you annotated, with the annotations you made, in JSON Lines format
- Look at each file in a text editor

Part six

- Convert the JSON Lines file to a CSV file
- Use the CSV file to calculate the average rating
- You may have some trouble extracting the scale
- I used `vi` to change the word `scale_1` to `scale`
- To do this to all five scale labels, I used the `vi` command `:%s/scale_[1-5]/scale/`
- Then I used the `extractIDandScale.py` script to extract the ID and scale from the JSON Lines file
- You may use an LLM for these tasks

Part seven

- I used `vi` to get rid of lines that start with *Politeness*
- I used the `vi` command `:g/^Politeness/d`
- Next I used the `calcAvg.py` script to calculate the average rating
- However, average ratings are not very useful for Likert scales
- Instead, I made a stem-and-leaf plot
- It took several tries for gpt-4o to make it like the one I can easily generate in R

Stem-and-leaf plot

```
5 | 000000
4 | 00000000000
3 | 0000000000000000
2 | 00000000000
1 | 00000
```

See if you can make a stem-and-leaf plot for your data with an LLM

END

References

Pei, Jiaxin, Aparna Ananthasubramaniam, Xingyao Wang, Naitian Zhou, Jackson Sargent, Apostolos Dedeloudis, and David Jurgens. 2023. “POTATO: The Portable Text Annotation Tool.” https://arxiv.org/abs/2212.08620.

Colophon

This slideshow was produced using `quarto`

Fonts are *Roboto Light*, *Roboto Bold*, and *JetBrains Mono Nerd Font*