Yield Strength

Week 11

Progress

- From last week: we found that GPT can generate more accurate yield strengths by feeding it some of the elements with values from our Ground Truth database.
- This week: We are going to expand the methods on training the data
- Issue from last week: GPT tends to generate values that I feed
    - Possible solution: give GPT proportion of Ground Truth Database as before, but then ask GPT one by one of elements' yield strengths that are not in the data I provided. Additionally, compare to the accuracy of those values before training and after training

Progress

- When asking GPT yield strength one by one, it will reach the rate limit

```
An error occurred 1th time: Rate limit reached for gpt-3.5-turbo-1106 in organization org-OuZk1sfr2wLCqczJdCqIRi9Z on tokens per min (TPM): Limit 60000, Used 5723
An error occurred 2th time: Rate limit reached for gpt-3.5-turbo-1106 in organization org-OuZk1sfr2wLCqczJdCqIRi9Z on tokens per min (TPM): Limit 60000, Used 5715
An error occurred 3th time: Rate limit reached for gpt-3.5-turbo-1106 in organization org-OuZk1sfr2wLCqczJdCqIRi9Z on tokens per min (TPM): Limit 60000, Used 5703
An error occurred 4th time: Rate limit reached for gpt-3.5-turbo-1106 in organization org-OuZk1sfr2wLCqczJdCqIRi9Z on tokens per min (TPM): Limit 60000, Used 5694
An error occurred 5th time: Rate limit reached for gpt-3.5-turbo-1106 in organization org-OuZk1sfr2wLCqczJdCqIRi9Z on tokens per min (TPM): Limit 60000, Used 5685
```

- In order to solve that, We add a try-except blocks that when such error occurs, we ask the program to sleep for 10 seconds and then rerun it

```python
try:
    sss.append({"role": "user", "content": qq})
    sss,ans = prompt(sss)
    sss.append({"role": "assistant", "content": ans})
except UnboundLocalError:
    time.sleep(10)
    sss.append({"role": "user", "content": qq})
    sss,ans = prompt(sss)
    sss.append({"role": "assistant", "content": ans})
```

Progress

We ask GPT yield strength value of materials in Ground Truth Database from index 100 - 159 one by one, total 60 materials. Below is the statistic before training:

```
Total: 60
Known: 49
# of correct: 14
Average precentage of error: 491.4523496155564
```

	Materials	Errors
0	Co25Ni25Fe25Al7.5Cu17.5	0.860724
1	(CoCrFeNi) HEA	0.620098
2	(FeCoNiCr)94Ti2Al4 HEA	0.116279
3	FeNiMnAlCr HEA (with 1.1 at. %% carbon)	0.126761
4	CoCrFeMnNi (larger grain size)	0.411765
5	tHEA-Mo	0.571429
6	Mo alloyed FeCoCrNi high entropy alloy	0.524390
7	CrCoNi (77 K)	0.678571
8	CrMnFeCoNi (77 K)	0.521739
9	CrCoNi (293 K)	0.333333
10	CrMnFeCoNi (293 K)	0.169811
11	AlCoCrFeNi	0.808000
12	Al0.3CuFeCrNi2	0.876000
13	Al0.3CuFeCrNi2 high entropy alloy	0.904000
14	Al0.7	0.933333

- Among 60 materials, GPT gets 14 of those with correct yield strength value.
- The reason that will the average percentage of error so big because there are two supper outlier: Cu and Zn

26	Cu	10999.000000
27	Zn	12999.000000

Progress

Below is the statistic after training

- Wrose known
- **Better number of correct 14 -> 27**
- Worse average percentage of error

```
Total: 60
Known: 35
# of correct: 27
Average precentage of error: 787.7167199711504
```

	Materials	Errors
0	Co25Ni25Fe25Al7.5Cu17.5	-1.000000
1	(CoCrFeNi) HEA	-1.000000
2	(FeCoNiCr)94Ti2Al4 HEA	-1.000000
3	FeNiMnAlCr HEA (with 1.1 at. %% carbon)	-1.000000
4	CoCrFeMnNi (larger grain size)	-1.000000
5	tHEA-Mo	-1.000000
6	Mo alloyed FeCoCrNi high entropy alloy	-1.000000
7	CrCoNi (77 K)	-1.000000
8	CrMnFeCoNi (77 K)	-1.000000
9	CrCoNi (293 K)	-1.000000
10	CrMnFeCoNi (293 K)	-1.000000

Summary

As we can see tell from the statistic in the previous slides, there is **trade-off** on whether to train GPT with Ground truth Database or not.

- Although # of correct yield strength materials increases (which is what what mostly care about), the number of known and average percentage of error decrease.

Questions:

- What will happen if we increase the train data size?
- What are some other add-ons that can minimize the trade off?

Additional Progress

- We also made a document in which we recorded our past methods and results for each one
  - https://docs.google.com/spreadsheets/d/1w1-PBFgJGAvK97znDEnTuV0-bch0S6JX0JdXxUXNKes/edit#gid=0
  - In this document we briefly describe the method, the number of generated values and the accuracy, along with other information

Hours Summary

Date	Hours	Description of Work
04/21/2024	2	Brainstormed ideas to further explore and how to implement training in a better way
04/22/2024	4	Adjusted training idea to only ask for and measure results for compounds not in training set
04/22/2024	1	Debugged rate limit
04/23/2024	2	Continued debugging and collected/analyzed results
04/23/2024	2	Compiled all work into slides
04/24/2024	2	Made document with methods and results