meeting 1 & meeting 2

**Next steps**

6.13 - 6.17 Finish the project plan and review the final version of my project plan

6.19 - 6.26 run the raw code and attempt the different methods (code implementation)

**Questions or things that might block the progress**

1. is there any similar literature that has implemented active learning on multi-classification problems from which the code can be reproduced

2. suggestions for project plans, ideas from other literature, and improvements

3. when PyCharm goes to implement the script, there is a default situation, prompting that the specified file cannot be found; I wonder what the problem is. Is this something that can be run directly with .sh?

4. if implemented, how long would the timeline be and how long would it normally take to complete the first version of the code?

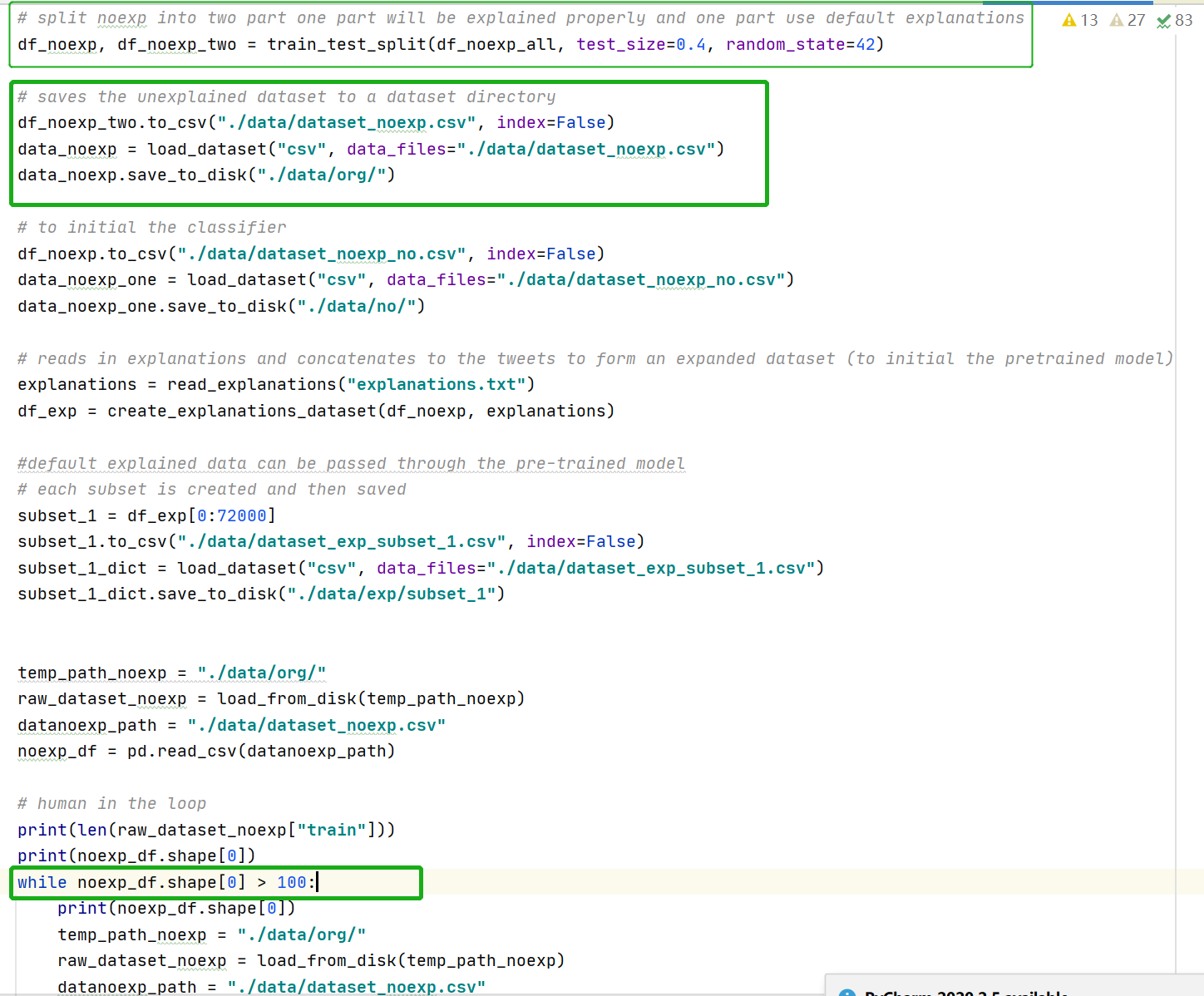
Meeting 3

**Update on what I have done so far**

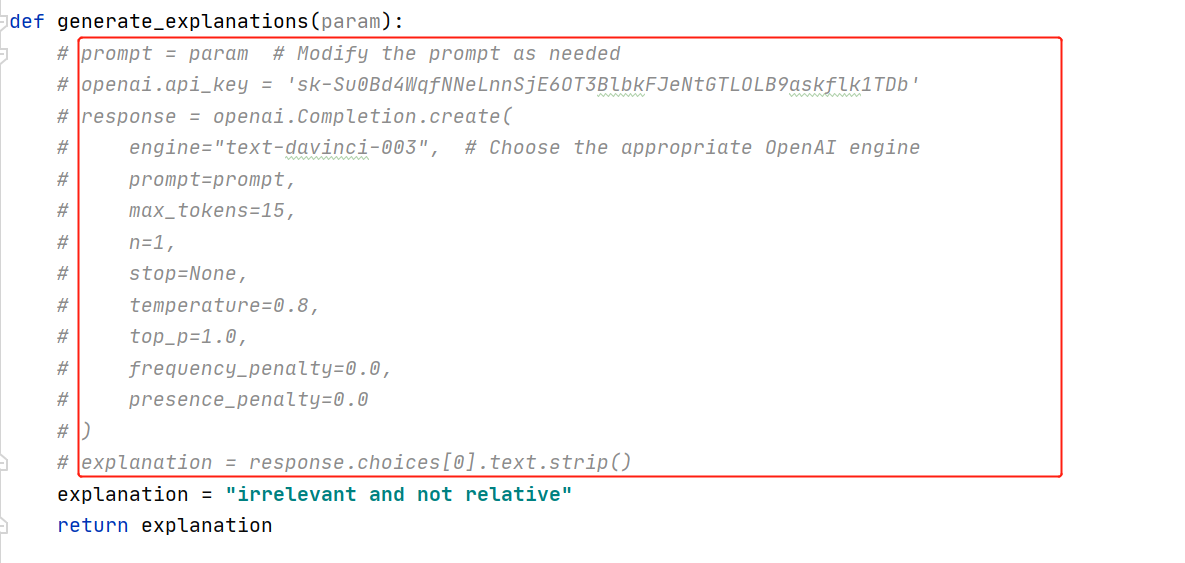
1. Update my project plan according to generate the explanation process and add ExpBERT and Openai API literature review.

2. Apply the openai API private key (30 reply cost 4 pounds)| Because it costs a lot, I use repeat words instead. I will use it when all the bugs have been repaired

3. Code implementation: Split no\_exp data into two parts and use random sampling to select the instance to give explanations. The explained data will be transferred to initialise the embedding process and classifier model.







1. Add explained samples to the default explanation dataset. Delete them from original dataset and repeat the steps.

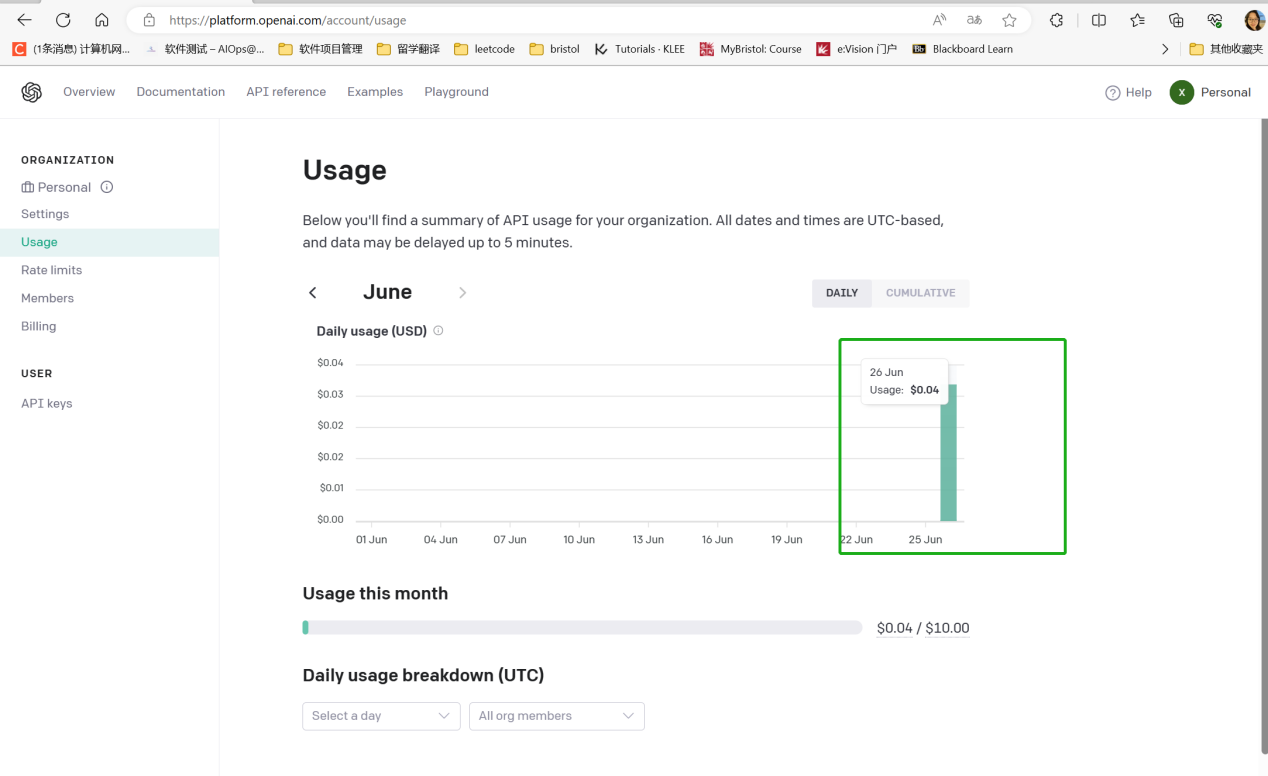




**Next steps**

1. Use uncertainty sampling to select samples instead the random sampling

2. Use a costless explanation generation method (low-cost openAI model)



3.Find how to evaluate the performance(the final test results as one loops’ performance)

**Questions or things that might block the progress**

1. Can I submit the project plan

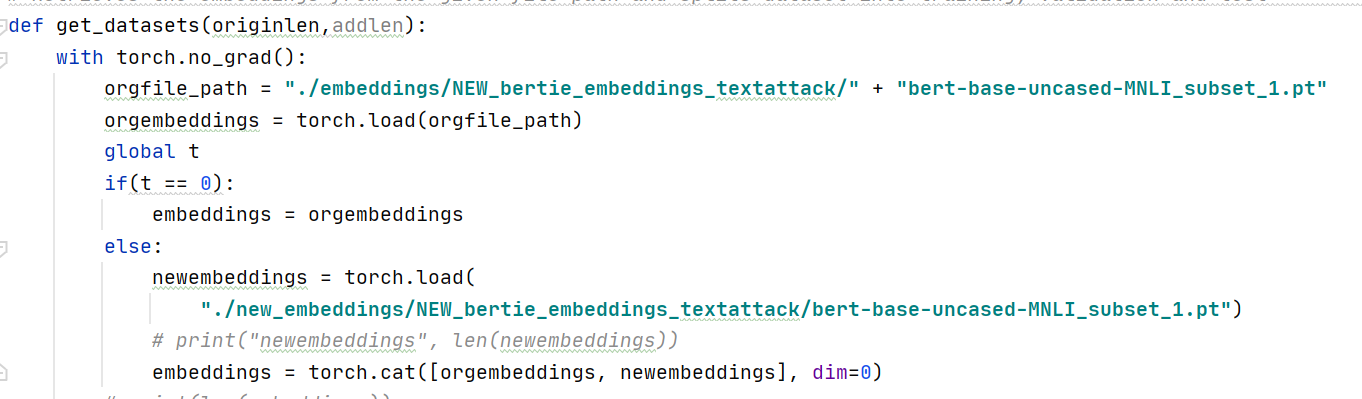
2. Is there any cost-friendly explanation generation methods? (openai is a little bit expensive)

3. Suggestion for the code implementation so far

Meeting 4

**What I have done so far:**

Because the newly explained dataset needs to be pretrained as well as the original default explained dataset, in case we pretrained the whole dataset in every loop, we pretrained the original dataset before the loop. In the loop, we only pretrained the new data we picked from the sampling process and contacted them to one embedding set and then put the embedding set into NN model.



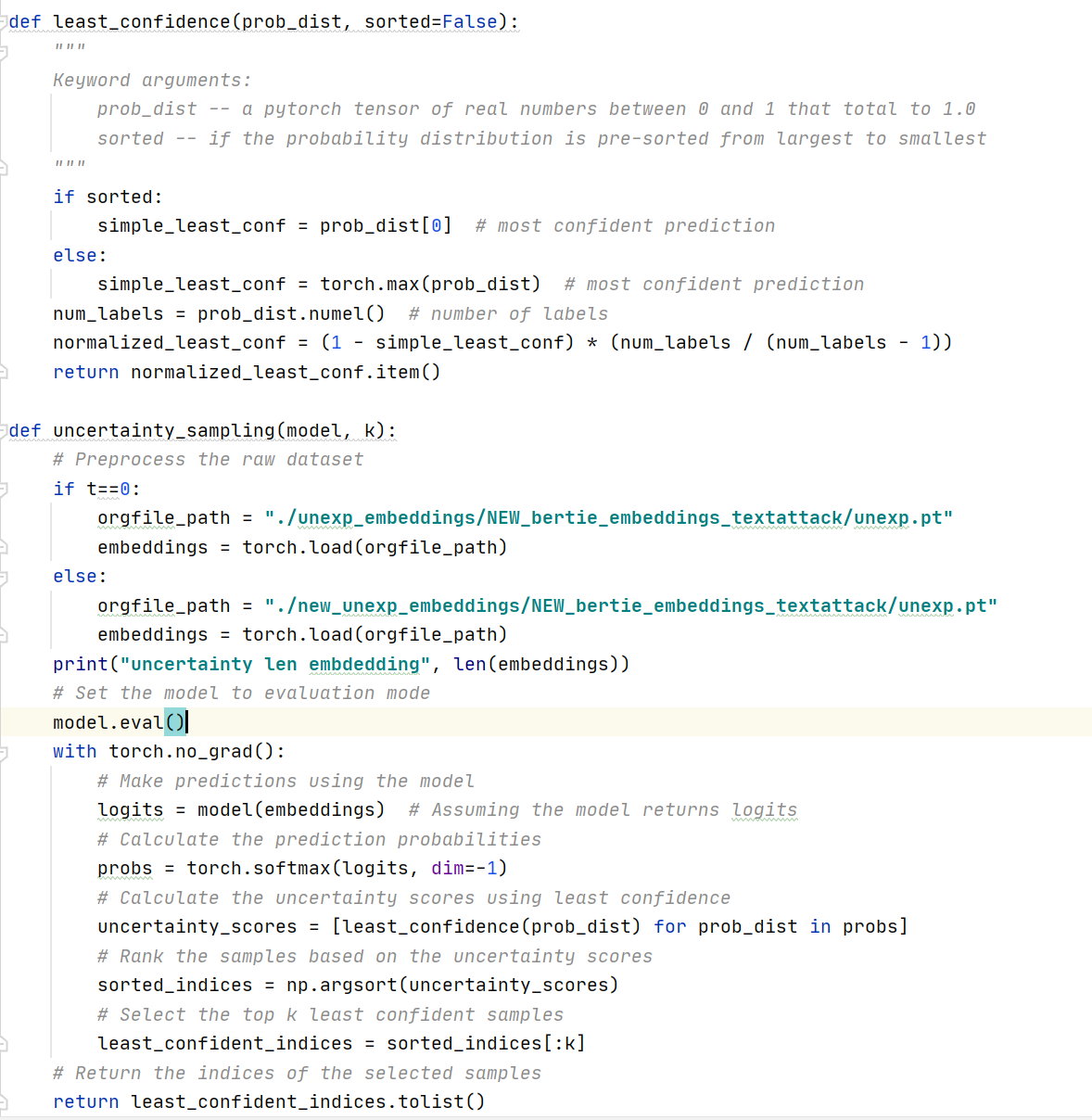
Load test dataset and pretrained before loop to reduce the time(Typhoon Philippine dataset)

Train and validation dataset pretrained before the loop (other datasets)



For uncertainty sampling:

Pretrained unexplained dataset with defualt explanation then put in the model to calculate the least confidence. After sampling process also delete the embedding set from origin pretrained unexplained embedding set.(reduce cost of time)



Also create training process without human in the loop

**Q**:

The default explanation set is good enough to catch the features, so the explanation generated by the Openai model may become a noise explanation (defining an accurate prompt is essential) or use a human annotator instead to monitor the best performance.

Using the NN model to calculate the uncertainty, the performance may not be significant compared with random sampling because the train data set is huge. In contrast, the new explained data is small(generate 50 samples a loop). So maybe need more loops.

**Next steps:**

1. Use a small amount of training dataset instead of a large one to pretrained and initialize the NN model(even the performance at the beginning is not good(40%), but after many loops, it may grow significantly(reach 60%+))
2. Diversity sampling and BALD acquisition function
3. Use a human annotator instead of openai model to find out what is the best explanation to improve the performance
4. Evaluate all functions (random sampling as baseline)

Note: