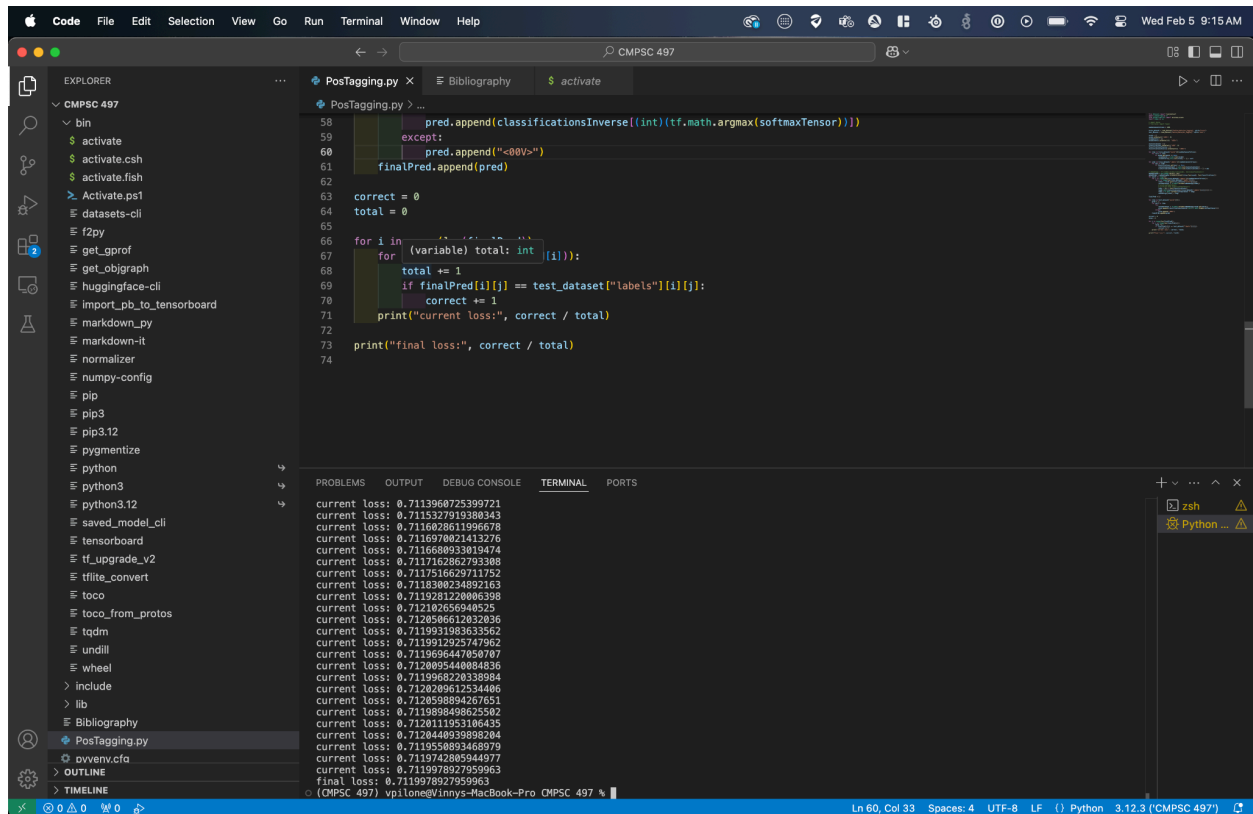


Below is a screenshot of my work, showing an over 71% accuracy when tested on all available testing data and trained on the first 1000 lines for one epoch. Please note that this was run in a local virtual python environment, so if there are any dependencies issues please feel free to reach out (I simply wanted to avoid uploading a truly ridiculous number of dependencies that I tried while experimenting with this HW).



```
58     pred.append(classificationsInverse([(int)(tf.math.argmax(softmaxTensor))]))
59     except:
60         pred.append("<@v>")
61     finalPred.append(pred)
62
63     correct = 0
64     total = 0
65
66     for i in range(100):
67         for (variable) total: int |i|):
68             total += 1
69             if finalPred[i][j] == test_dataset["labels"][i][j]:
70                 correct += 1
71         print("current loss:", correct / total)
72
73     print("final loss:", correct / total)
74
```

```
current loss: 0.7113660725399721
current loss: 0.7115327919388343
current loss: 0.7116828611996678
current loss: 0.7116978021413276
current loss: 0.7116688933919474
current loss: 0.7117162862793308
current loss: 0.7117516629711752
current loss: 0.7118380234892163
current loss: 0.7119261220806398
current loss: 0.712182656940525
current loss: 0.7120586612032036
current loss: 0.711931983633562
current loss: 0.7119912925747602
current loss: 0.7119696447050707
current loss: 0.7120895440884836
current loss: 0.7119968220338884
current loss: 0.7120289612534406
current loss: 0.7120598894267651
current loss: 0.711989498625502
current loss: 0.7120111953106435
current loss: 0.7120440939898204
current loss: 0.7119550893468979
current loss: 0.7119742805944977
current loss: 0.7119978927959963
final loss: 0.7119978927959963
```

Additionally, I did my best to keep a bibliography of all referenced resources, so if there are any concerns as to where I got particular code ideas or algorithms please feel free to consult that or reach out to me directly. Thanks!