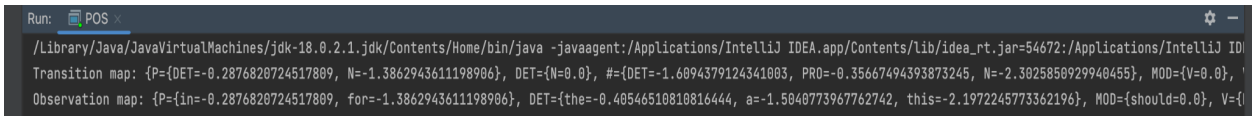
Training:

- In this first phase of training, we used data from the simple-train-tags and simple-train-sentences files to train our program. To do so we created a ***transitionsMap*** - stores the transitions and their probabilities, and ***observationsMap*** - stores the observations data that's words with their tags and their log probabilities based on frequencies. Here is the output of the transitions and observations map.



- After training our program we used the actual **simple test files** with the viterbi algorithm. For the viterbi algorithm, we used the pseudocode from the course website and backtracked the maps in the arraylist of maps to get and compared the returned list of tags to the ones read from the tags file.
- Based on this, our calculated the percentage accuracy of the program is approximately 86%

- To test the brown files, we trained our program using the **brown-train-tags** and **brown-train-sentences** files and tested with the actual files **brown-simple-tags** and **brown -simple-sentences.** Using the viterbi algorithm, we found the percentage accuracy to be approximately 95.7%.
- Next, we hard coded the training data using PD7 training data to test if our program is working well. The results are as follows:

3 Training

Listing 1: Training Data

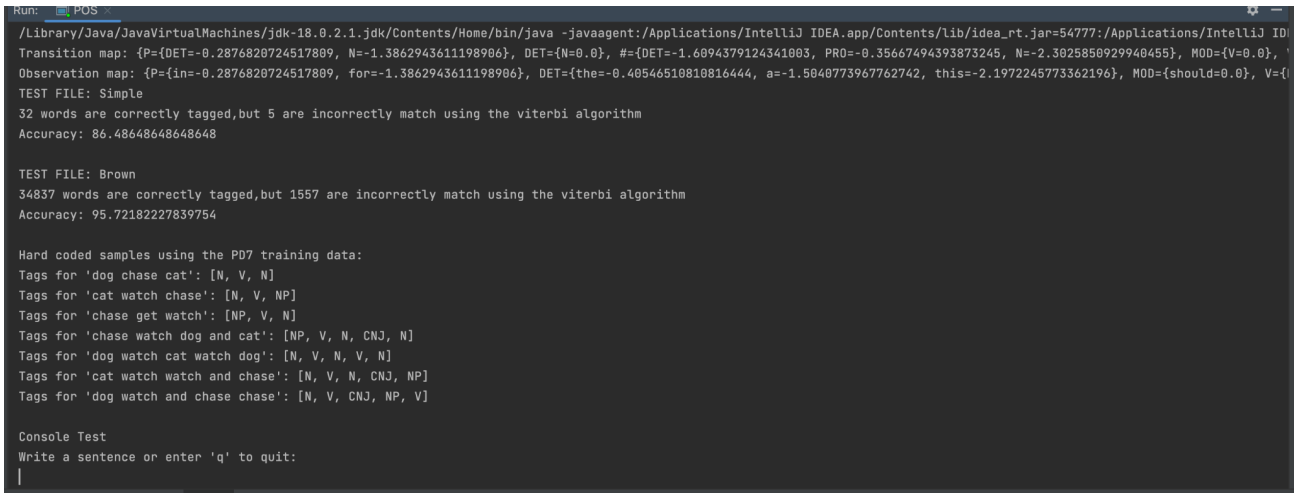
```
1 cat/N chase/V dog/N
2 cat/N watch/V chase/NP
3 chase/NP get/V watch/N
4 chase/NP watch/V dog/N and/CNJ cat/N
5 dog/N watch/V cat/N watch/V dog/N
6 cat/N watch/V watch/N and/CNJ chase/NP
7 dog/N watch/V and/CNJ chase/V chase/NP
```

3.1 Emission map

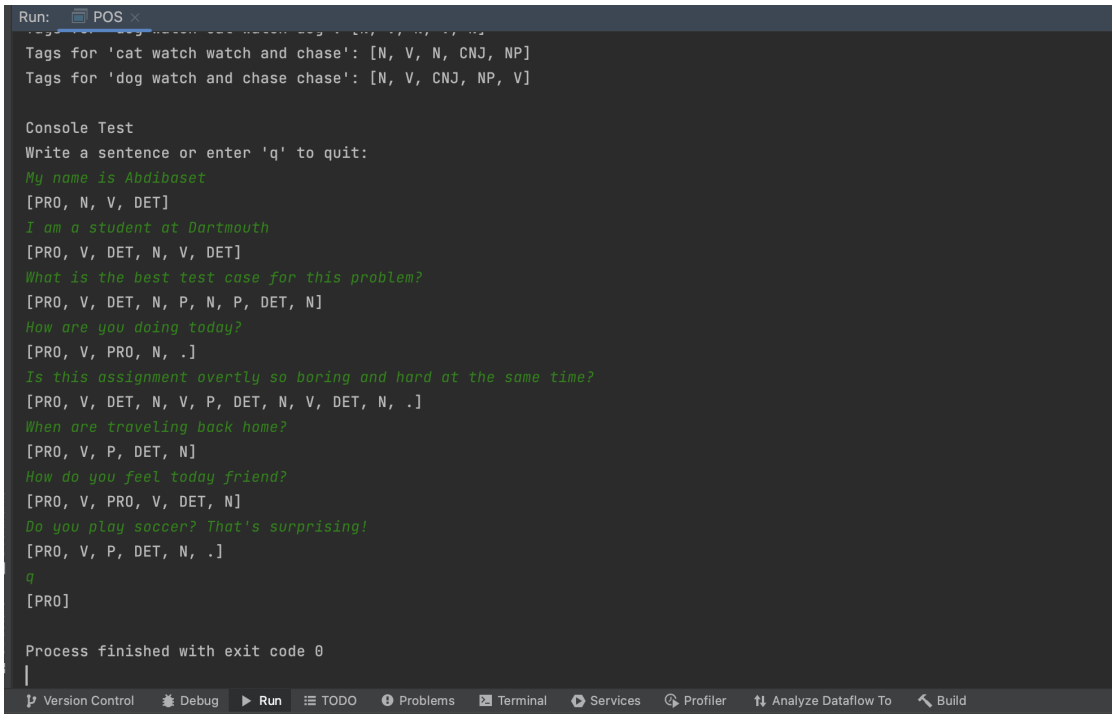
tag \word	and	cat	chase	dog	get	watch	normalize by
CNJ	3	0	0	0	0	0	3
N	0	5	0	5	0	2	12
NP	0	0	5	0	0	0	5
V	0	0	2	0	1	6	9

3.2 Transmission map

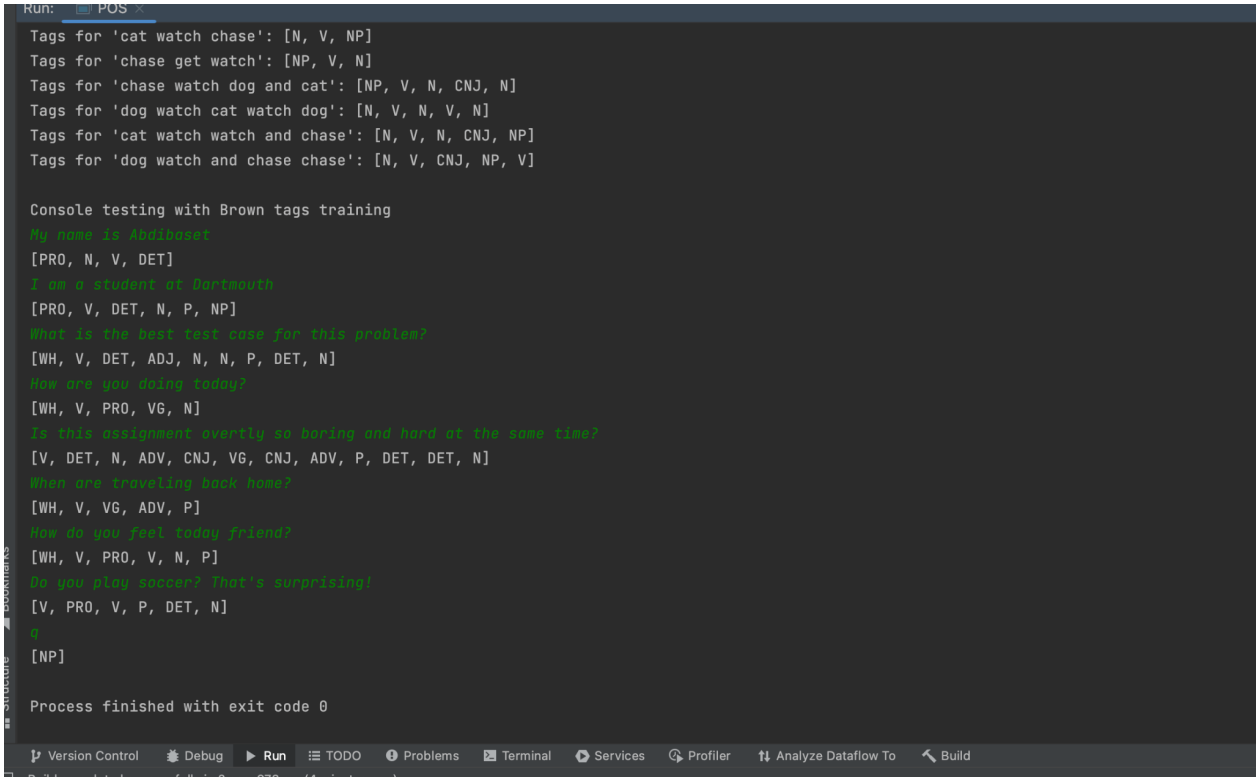
From\To	CNJ	N	NP	V	normalize by
#	0	5	2	0	7
CNJ	0	1	1	1	3
N	2	0	0	6	8
NP	0	0	0	2	2
V	1	6	2	0	9



- Lastly, we did a console test using the train data from the **simple-train-tags** and **simple-train-sentences that** we found to be at least 86% accurate.



- We also did console testing using the trained data from **brown-train-tags and brown-training-sentence** and here is the output for the same sentences. The training with **brown train data** gives more accurate tagging because it is a large data set.



-