Part of speech tagger

Please note that this code uses Python 3.

To run this, type: `python main.py`

In the `main.py` file, there are several variables, which can be changed to train, test, and output different files.

The tagger is located in `tagger.py`.

The program will also produce output for edge case sentences, and print all caught OOV words for debugging purposes. These can be toggled with the `debug` boolean in `tagger.py`.

How it works

The data is read in and parsed to create a table of frequencies (provided by `frequency_table`), which also outputs the probabilities. The frequency table is a utility class over a table of tables, and is used to hold the bigram transition frequencies, as well as the word given part of speech tag frequencies.

A tagger class implements a sentence level Viterbi algorithm, given these two frequency tables, and also handles edge cases such as OOV words and 0 probability sentences.

OOV Word handling

The OOV word handler makes several assumptions:

- words containing numbers have 0.75 probability of being CN (cardinal number)
- words with dashes have 0.75 probability of being JJ (adjective)
- capitalized words without ending in "s" have 0.75 probability of being NNP (proper noun)
- capitalized words ending in "s" have a 0.75 probability of being NNPS (proper noun plural)
- all other OOV words have a probability of being a POS tag based on the probability of a tag occurring randomly, ignoring any emission probability

Performance

Trained on the `WSJ_02-21.pos` corpus, the file performs ~94% accurancy on the `WSJ_24.words` file given the corresponding `.pos` file as key.

However, the HMM did run into a few cases where zero probability occurred for some unknown reason (2 sentences in `WSJ_24.words`), possibly simply by chance or an unhandled edge case in the HMM. For these cases, the tagger will produce tags based on the emission frequency of the tag given the words as a heuristic. Talking to the professor, it seems like an unhandled edge cases is more likely, despite the fact that the tagger already accounted for OOV words (could be an issue with overflow/underflow with small numbers).