

Gemini wrote me a small script to show how many possible predictions in the ChatGPT-2 model: 50,257. Numbers farther from 0 are less and less likely to be predicted by the model. And of course a value of x=0, the most likely prediction, does not produce the original poem – it shows what it thinks is possible based on the prompt of the single line of poetry. Not a single final word matches the original even at x=0; it's extraordinarily unlikely that any of the predictions would exactly match the original poem itself. I wonder how likely it is to get even a single word correct, given the nature of poetry and the nature of computational prediction! Larger numbers produce things utterly lacking in cohesion – numbers, symbols, words in other languages, while smaller numbers make relative sense given the sentence.

I thought it would be very tricky to figure out how to extract every noun, but turns out it's really easy, and something regularly done with the Natural Language Toolkit (NLTK)! Each word (every word? seems impossible?) is tagged with something indicating its part of speech - for instance, NN or NNS are the tags for singular and plural nouns, respectively. Each word is stored as a tuple containing the word itself and its tag. So then it would be fairly easy to do the same sort of extract/replace method as done with the last word in the line, it's a matter of parsing and replacing with the same method of measuring the 7th most likely predicted word as used in the rest of the assignment.

```
tagged_words = nltk.pos_tag(words_in_line)

new_line_words = []
for word, tag in tagged_words:
    if tag.startswith('N'): # Check if the tag indicates a noun (NN, NNS, NNP, NNPS)
        new_line_words.append('[REPLACED_NOUN]') # Replace noun with a placeholder
    else:
        new_line_words.append(word)
```