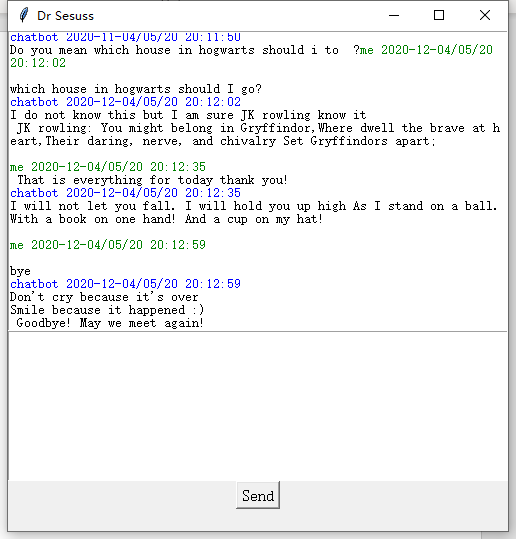
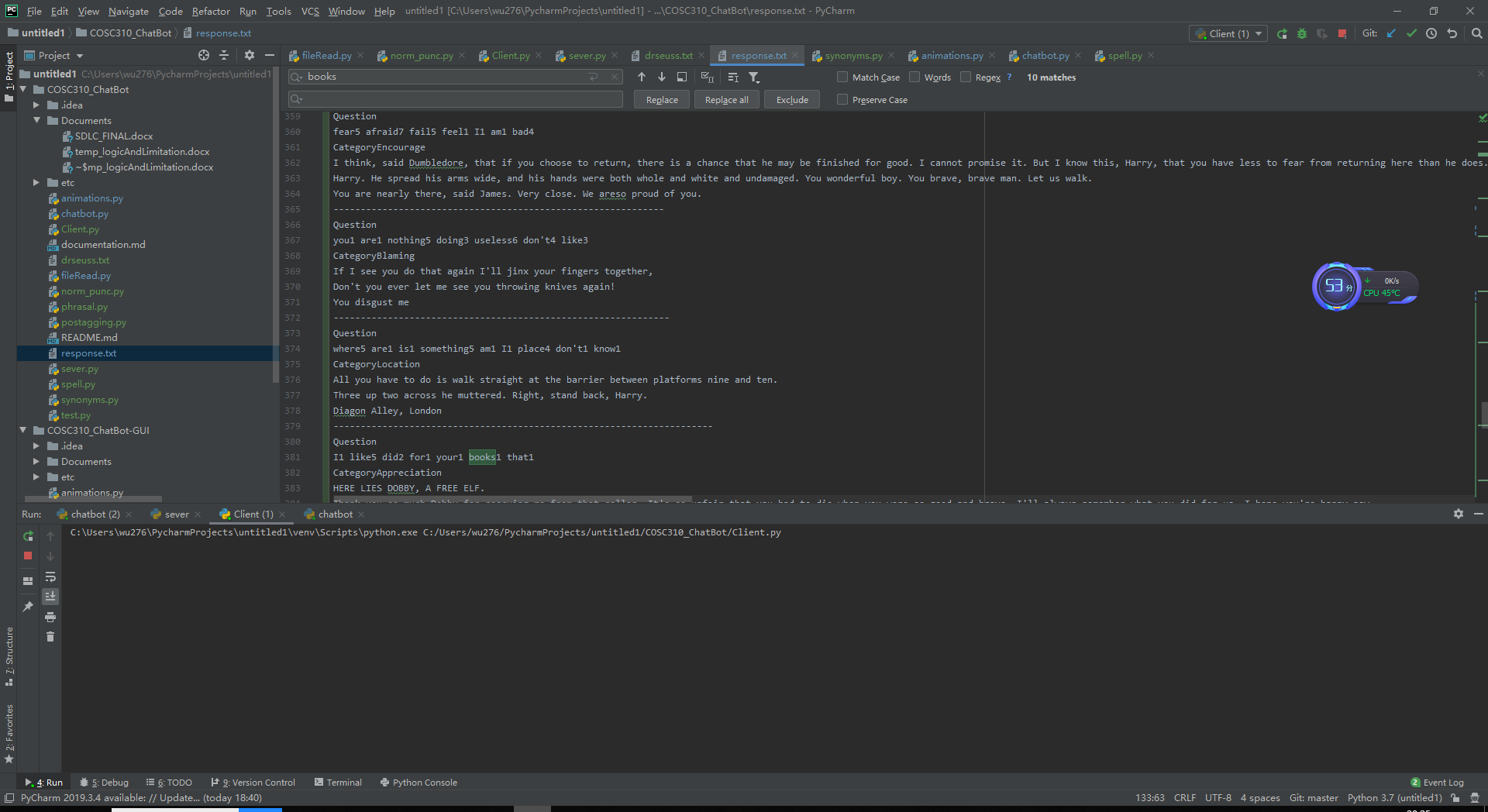
**GUI:**



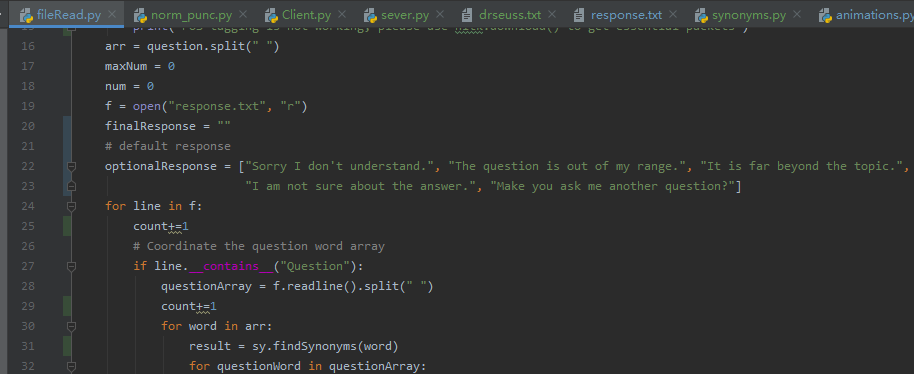
The GUI will looks like this, with a char screen and a input frame. You can click send or enter to chat with the chat bot via socket.

**Second topic:**



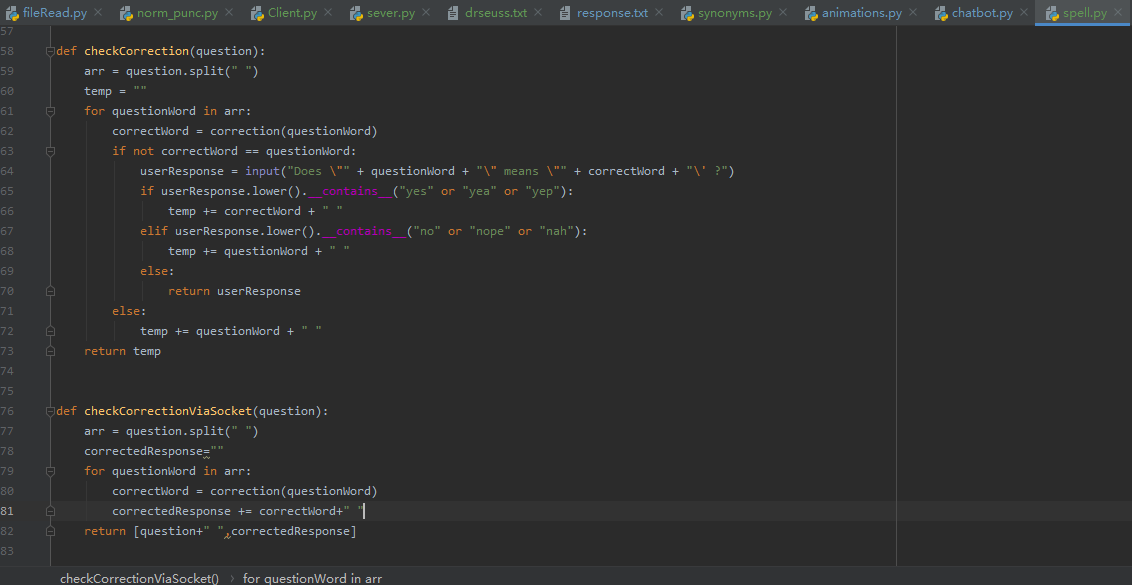
For this response.txt file, it includes the topic of two authors, one is dr.Sesuss and the other one is JK rowling. The topics before line 227 is dr.Sesuss’ topic and the after that will be JK rowlings.

At lease 5 response for outside topic response

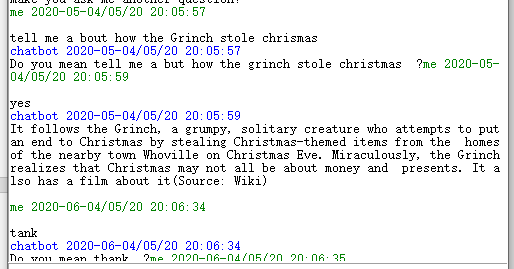


The default optional response has a length of 5, which shows that we have 5 outside topic response.

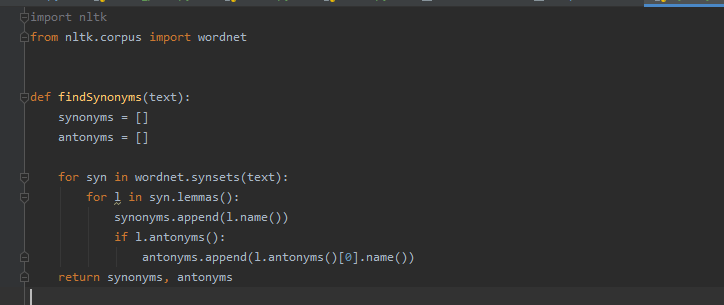
Spelling mistake handling

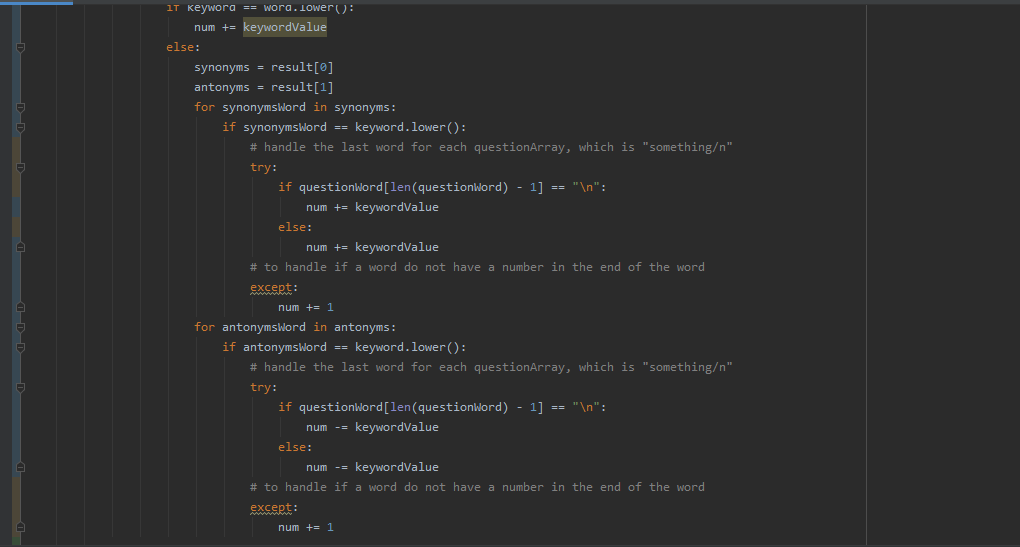


The spell.py has the method to correct the spelling mistake. The basic idea is to look up a big file and look for and return all the words that need to edit 0, 1 or 2 character, and then find the word that has the maximum similarity compare with the original word. The check Correction will ask user if the word is he want. In socket will look like this.



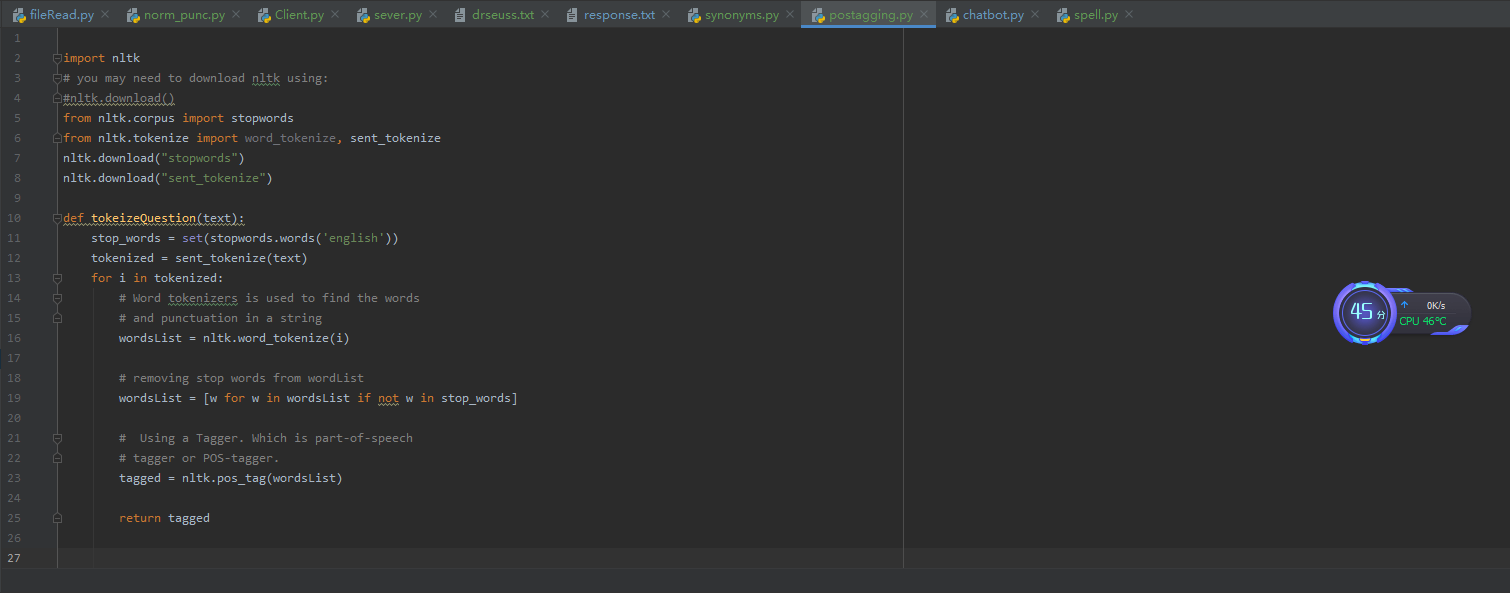
Synonym recognition:

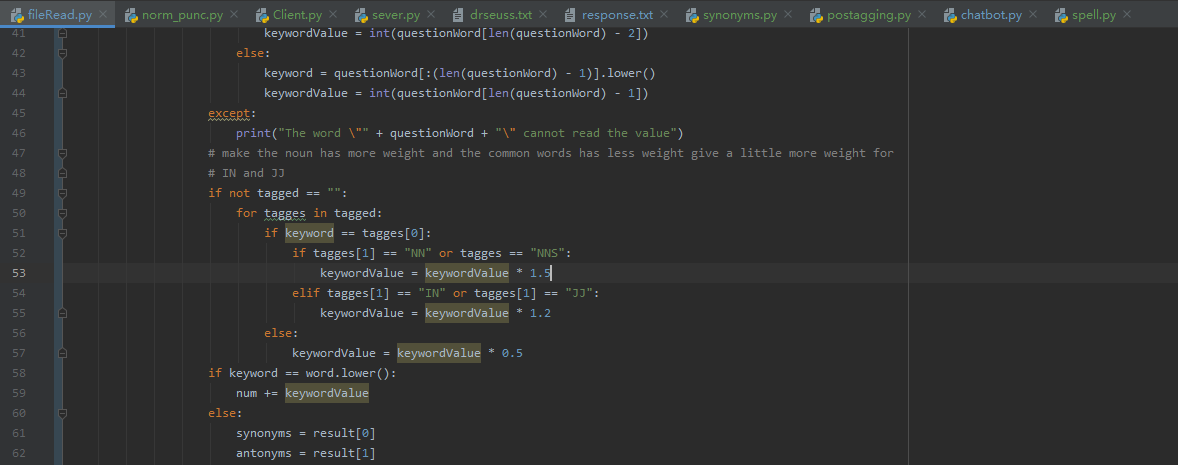




The first picture is from synonums.py, it imports nature language toolkit, that can find all the synonyms and antonyms the words. In fileRead.py in the second picture, When input a question, if the question word does not match the key word. Then it will find all synonyms of this word and see if these words match the key words. If a synonym matches a key word, add the value. If an antonym matches, subtract the value of the keyword.

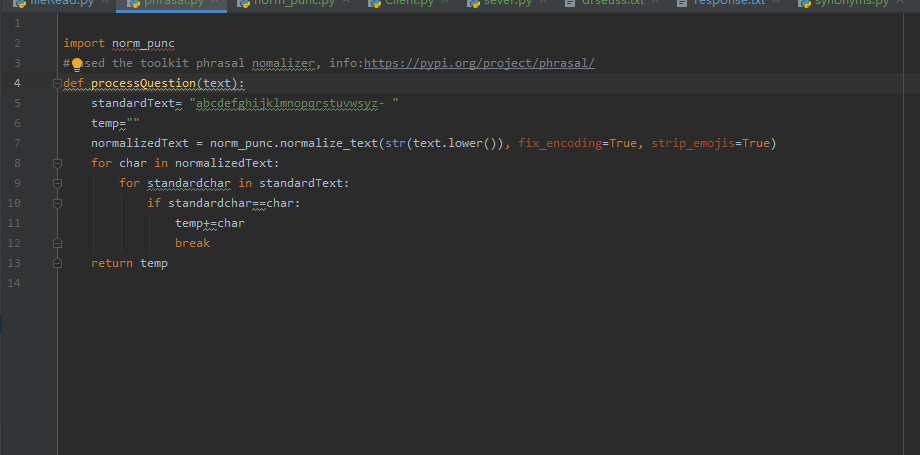
POS tagging





The first picture is from posstaging.py, it imports nature language toolkit, that can find all the tags of the words. In fileRead.py in the second picture, when inputting a question, posstaging.py will find the tag of each question words. Whenever the question word matches, it will go find to look up its tag. If the question word is noun we give it a larger scaler for calculation, and give no tag word smaller scaler since they usually are common used words. We also give other tagged words different scaler, you can check in the file.

Phrasal



The phrasal.py shows how we use normalizer of phrasal to process the word, firstly we use normalize text to normalize spaces, replace combining diacritics and normalize dashes and quote marks. Also, we delete all useless symbols for the question, and transform the question to be a perfect format for further using by spell correction and check question function.

Socket：

The socket is built in sever.py and client.py host in local host. Both Sever and client are able to send and receive message when connection has established. When sever is on, if User want to chat with the chatbot then, he can run the client.py and a screen will shown up. The conversation will be made via socket.

