Method used for creating a corruption data (file: corruption_data.py)

These are the three possibilities I have considered to change in a given sentence at the same time to challenge my system built in (corruption_classifier.py)

1) Change a word to another from wordnet that starts with the same letter and has the same number of characters.

2) Randomly misspell words by changing the last letter of a word to a random character from the alphabet.

3) Randomly removing at max 1 word from the sentence.

I have two functions, one for picking a random word from starting with a given character and the length from wordnet which I have imported from the nltk.corpus. I also have a function that gives a sentence the corruption of that sentence taking into account the above three possibilities. A brief description of this function: I use an averaged_perceptron_tagger on the tokenized sentence and based on a random generated probability I will decide to remove that word, misspell it, change the word or do nothing to it, this is how a new corrupted sentence for a given sentence is generated. One main thing that I have made sure of is that recursively calling the function to corrupt the original sentence if the generated corrupted sentence is the same as the original sentence. Also, I have taken into consideration that even if the difference between the original and the corrupted sentence is just a space, then again recursively call the function to get a proper corrupted sentence which follows at least one of the three steps that I have stated above.