Welcome to report 2

Exercise 2

Name: Hala Grefat

Id: 207931008

Part1:

in this part I got through the text and calculated the probability for each word in unigrams, for each two words and in Trigrams for each three.

The functions that calculate probability: *uni_grams_prob, Bi_grams_prob* and *Tri+grams_prob.*

For both uni_grams_prob, Bi_grams_prob calculating the probability was with Laplace smoothing, giving the word <unk> the prob of a word that isn't in the text.

For Trigrams, calculating the prob for each three words/tokens was without smoothing.

After having the probabilities all calculated, in functions: Classify_Uni, Classify_Bi and Classify_Tri I calculated the probability for each line in phrases, according to the probabilities for the corpus, and for words that aren't in corpus It had the prob of <unk>.

At this part of the process came the Backoff Linear Interpolation for Trigrams, with parameters:0.6, 0.35, 0.25 for Trigrams, Bigrams and Unigrams accordingly.

My reasoning for choosing these parameters is that I wanted to give the biggest weight for trigrams and bigrams, because although the sparsity of three\two words sentences can give lower to zero probability to sentences, they are more accurate and can give better information about the text.

Part2:

In this part, using the same functions I used before to calculate the probability of corpus, except this time without smoothing (as instructed in the hw)

In function *generate_text* given a random length from corpus's sentences lengths, It generates a sentence according to one of the models (Unigrams,

Bigrams, Trigrams). I used function random.choices () that takes a list of words to choose from and a list of weights for each word, for Bigrams and Bigrams I looked for the relevant words(pairs and trios) according to previous token\pair and send those to choose from with their weights, since they are more probable.

Answers to the questions:

1. I could tell from the output that, sentences that were generated by Unigram model, didn't make sense, and were basically a bunch of random words that came up a lot in the corpus, in comparison sentences that were generated by Trigrams made sense and were even perfect at times, Bigrams were in between.
2. Unigrams, because it chooses words with higher probability randomly, regardless of previous words.
3. Yes it did switch languages, for unigram model and bigrams model I don't think we can consider it a code switch, because the way they work doesn't give too much consideration to the context of the sentence, for Trigrams although this model does give more weight to the previous words and so to the context, the fact that we merged the files means that they aren't really related and so they shouldn't be in the same sentence as a code-switch, which explains why it was more rare to have multiple languages in the same sentences that were generated by Trigrams.

Part3:

1. Example: you're right away. [you're right] [right away]
2. Example: I've a problem occurred [I've a problem] [a problem occurred](omg this took so long)
3. The second one, with three words was more difficult to come up with, because for the second group of three words to be Grammarly correct, there aren't many options that don't make sense with the previous set of words, this suggests that language isn't very "creative" which was something that was said in the lecture.

I felt like I was racing with time doing this hw, I hope I didn't forget to do anything lol. Have a nice day 😊

PS: I resubmitted the hw, because the code was **very** slow.