Assignment 3 (CSC3160)

The total score is 100 marks and it will be converted into **8%** for your final grades. Please use the following sentences for this assignment.

- Sentence 1: *Tonight, I will make the evening meal.*
- Sentence 2: *I am going to make dinner tonight.*

Task 1: Language Models [30 marks]

1. Implement an N-gram language model (e.g., bigram or trigram) using the four provided text files (`country.txt`, `pop.txt`, `rap.txt`, and `rock.txt`).
2. Predict the next word of each n-gram of the words in Sentence 1 using your N-gram model. For example, what is the predicted next word given two words "Tonight I" using the trigram model. You can answer this question in the following format.

```
Input: (Tonight, I) --- Output: {the_predicted_text}
Input: (I, will) --- Output: {the_predicted_text}
Input: (will, make) --- Output: {the_predicted_text}
...
```

Task 2: Word Embeddings [70 marks]

1. Use a pretrained word2vec model to conduct some experiments. Please refer to the hints[1][2]. Please load the pretrained word2vec model with the following codes.

```
!pip -qq install gensim
import gensim.downloader as api
model = api.load('word2vec-google-news-300')
```

   (a) Get the five most similar words to "speech".

   (b) Confirm that "$queen''$ = "$king''$ − "$male''$ + "$female''$ ("$queen''$ should be the three most similar words of the right-hand equation.).

2. Calculate the similarity of two sentences using the word2vec model.

[1] Gensim Tutorials: https://radimrehurek.com/gensim/models/word2vec.html
[2] Gensim Documentation: https://tedboy.github.io/nlps/api_gensim.html

3. The function below is called Jaccard similarity. Explain how Jaccard similarity computes the similarity of sentences in a few sentences. And, calculate the Jaccard similarity of Sentence 1 and Sentence 2.

```
def jaccard_similarity(sentence1, sentence2):
    # sentence1 and sentence2 are the strings.
    tokens1 = set(sentence1.lower().split())
    tokens2 = set(sentence2.lower().split())
    intersection = tokens1.intersection(tokens2)
    union = tokens1.union(tokens2)
    return len(intersection) / len(union)
```

4. Why do the similarity scores of word2vec and Jaccard similarity differ a lot?

You may use libraries such as NLTK for N-gram modeling and Gensim for word2vec. Refer to the lecture notes on language models and word embeddings