THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

COMP 4221 Introduction to Natural Language Processing

Course Project

Name: Lam Leung Kin

Student ID: 20862105

Instructions to students:

1. The course project contains ***3*** tasks. You need to finish all of them.

2. You need to upload the required files on Canvas. Some of the tasks require you to show some results. You may report the results in this document and upload it together with other files on Canvas.

3. Please note that there is no requirement on the quality of the natural language generation.

4. Please don't forget to write down your name and student ID.

Task 1. [Byte-Pair Encoding (BPE)**]** Different from white-space segmentation and single-character segmentation, Byte-Pair Encoding (BPE) is a data-driven text tokenization algorithm. We consider using the corpus "text_news.txt" given on Canvas**.** **(30 Points)**

Please read the following Github page of the tool "SentencePiece":

https://github.com/google/sentencepiece

(a) Please train BPE learner on the corpus "text_news.txt" by using SentencePiece. The number of merges should be (k+1) * 100, where k is the last digit in your student ID (e.g., if your

student ID 2987876, then the last digit k in your student ID is 6). You need to upload the .model file and the .vocab file. **(10 points)**

(b) Please use the BPE segmenter to tokenize your name (which should be the same as the one on your student ID card) and show the tokenization result. **(10 points)**

(c) Please apply the BPE segmenter to "text_news.txt" and upload the tokenized file. **(10 points)**

Task 2. [N-Gram Language Model] Language Model is one of the most important NLP models. N-Gram Language Model is the de-facto language model before the age of deep learning. We will also use the corpus "text_news.txt" which was uploaded on Canvas. Please read the source code "trigram.py" in Python which trains trigrams on the corpus "text_news.txt" and generates text based on the trigram. Please put the source code together with the corpus under the current folder. You need to install NLTK (https://www.nltk.org/install.html) in this task and please follow the instructions on the website. **(50 points)**

i. Please run the source code "trigram.py" and show the text generated by the code. **(10 points)**

ii. The source code "trigram.py" is an implementation of trigram. Please modify it to make it an implementation of four-gram and train the four-gram on "text_news.txt". Please upload the modified code. Please also generate (k+1) * 5 new words based on the context "It was just a normal summer day when Allan Houston stopped his workout in midchurn" and show the results of the generation. Note that k is the last digit in your student ID (e.g., if your student ID 2987876, then the last digit k in your student ID is 6). **(20 points)**
Hint: Please refer to the NLTK Ngram toolkit for the guideline. (https://tedboy.github.io/nlps/generated/generated/nltk.ngrams.html)

iii. In the implementation of trigram in "trigram.py", it uses the random sampling in the next word prediction (that is, when we predict the next word followed by the two words a and b, we random sample one trigram <a, b, c> from all trigrams and output c as the next word). Please modify it to make it an implementation of greedy algorithm for the

next word prediction (that is, when we predict the next word followed by the two words a and b, we find trigram <a, b, c> from all trigrams with the highest frequency and output c as the next word). Please upload the modified code on Canvas. Please also train the trigram on “text_news.txt” and then generate (k+1) * 5 new words based on the context "It was just a normal summer day when Allan Houston stopped his workout in midchurn" by using the greedy algorithm and show the results of the generation. Note that k is the last digit in your student ID (e.g., if your student ID 2987876, then the last digit k in your student ID is 6). **(20 points)**

Task 3. [Neural Language Model with LSTM] Language Model is one of the most important NLP models. Neural Language Model is the de-facto language model in the age of deep learning. We will adopt the corpus “sample.txt”. Please read the source code “lstm.py” and “lstm-text.py” in Python which trains LSTM-based neural language model on the corpus “sample.txt” and generates text based with the LSTM trained. Please put the source code together with the corpus under the current folder. You need to install TensorFlow in this task and please follow the instructions on their website (https://www.tensorflow.org/install). **(20 points)**

(a) Please run “lstm.py” to train the LSTM model and show the text generated in the last epoch. **(5 points)**

(b) Please run “lstm-text.py” to generate text and show the text generated. **(5 points)**

(c) Please modify the “lstm-text.py” to make it generate 200 new characters conditioned on the the context " The United States used another comeback yesterday to take". Please upload the modified code and show the text generated.
(10 points)