Programming: Language Identification

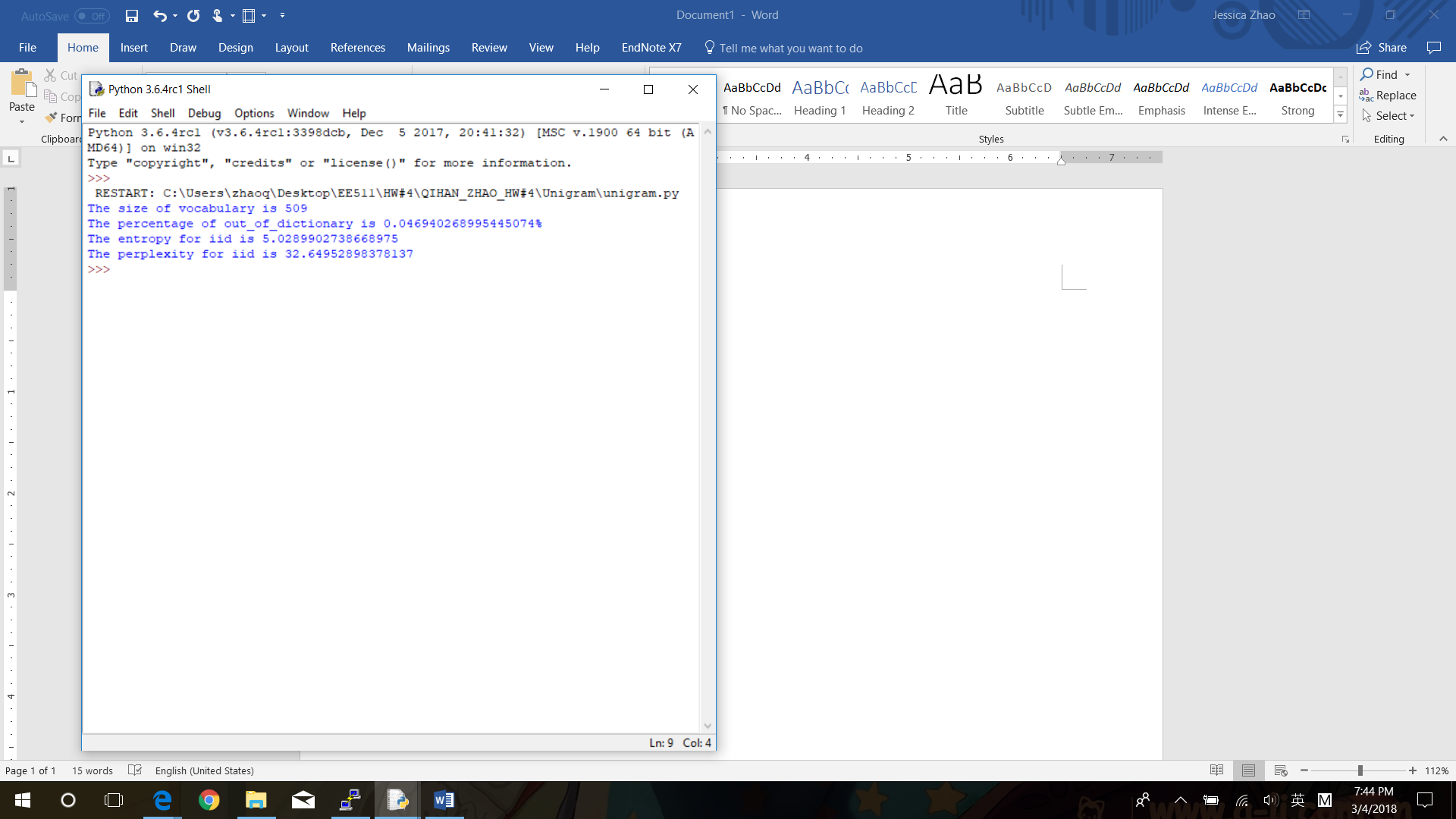
1. Warm-up

Size of vocabulary: 509

Percentage of out of vocabulary tokens: 0.0469%

Entropy for validation data is: 5.03

Perplexity for validation data is: 32.65



1. Recurrent Neural Network

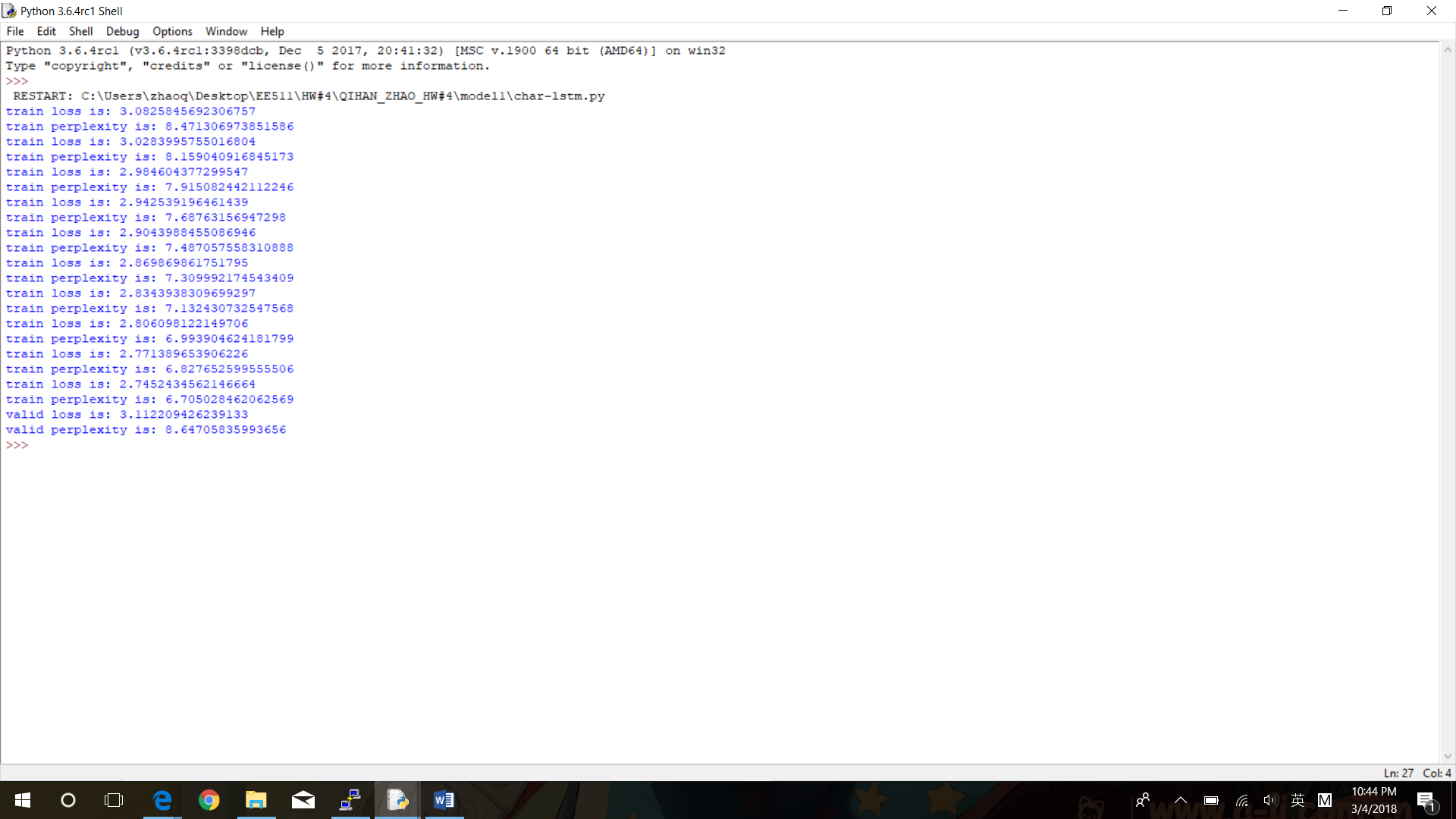
ReadMe document explains how to run the code inside folder named as model1

The model is character-based LSTM model, which embeds input characters and outputs predictions for next character.

After training the model and validating on the validation data,

the entropy for validation data is: 3.11

the perplexity for validation data is: 8.65



1. Language identification

ReadMe document explains how to run the code inside folder named as model2

The model is also character-based LSTM model, but it embeds both input characters associated with the language id. It takes time to think about how to feed both a character and a language embedding and what is the output from RNN hidden layer at this time. I find the main idea is that based on given labels, the probability on predicting next characters varies. In the other word, a correctly assigned label will maximize the accuracy to predict next character and minimize the cross entropy. Based on designed model and feed both character associated with language embedding, we assign the text with the label minimizing the cross-entropy. However, the trained model does not work very well for the validation set.

The test.ids.csv is in the folder named as model2, which produced by test.py.