# Bigram/ Trigram language model for word sequences

## Introduction

The purpose of this document is to report the outcomes for the 1st Assignment of class Text Analytics, April 2020. Code for the assignment can be found by following the link: [www.zouzounompalaxki.com](http://www.zouzounompalaxki.com)

## Corpus Preprocessing

The corpus used in the assignment is “The parent's assistant” by author Maria Edgeworth and was downloaded from NLTK Python Package.

For the preprocessing of the corpus three methods were implemented and used:

1. special\_char\_remove(corpus: str) -> str

Gets a string as input and gives a string consisting of letters and the characters “.,?,!” as output.

1. sentence\_tokenization(text:str) -> list

Gets a string as input and gives a list of sentences as output. The nltk.sent\_tokenize() method was used for the sentence tokenization.

1. word\_tokenization(text:str) -> list

Gets a string as input and gives a list of words as output. The nltk.word\_tokenize() method was used for the word tokenization. The method also removes specific contractions by looking for them in a dictionary and substituting them with the full word.

The corpus was processed by the three methods in the order given above, leaving a list of 10,168 list/sentences and a total of 898,088 words (characters “.,?,!” included).

Figure 1: Part of corpus before preprocessing

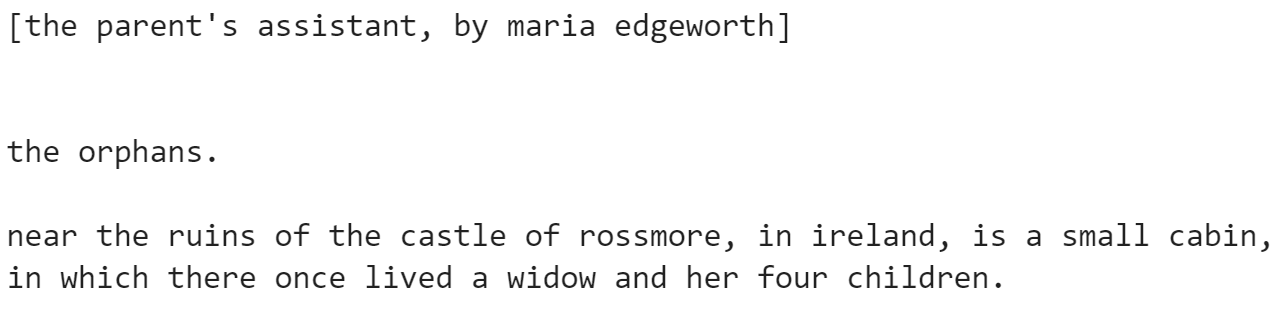
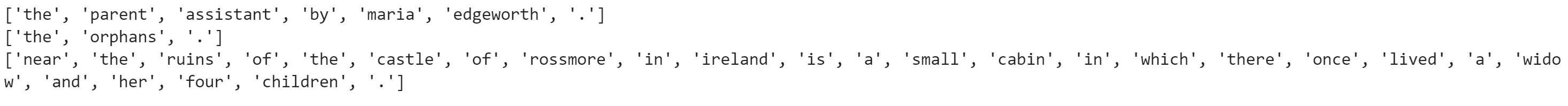


Figure 2: Same part as "Figure 1" after preprocessing



As a final step, the corpus was split to Train, Development and Test set according to 60%-20%-20% rule.

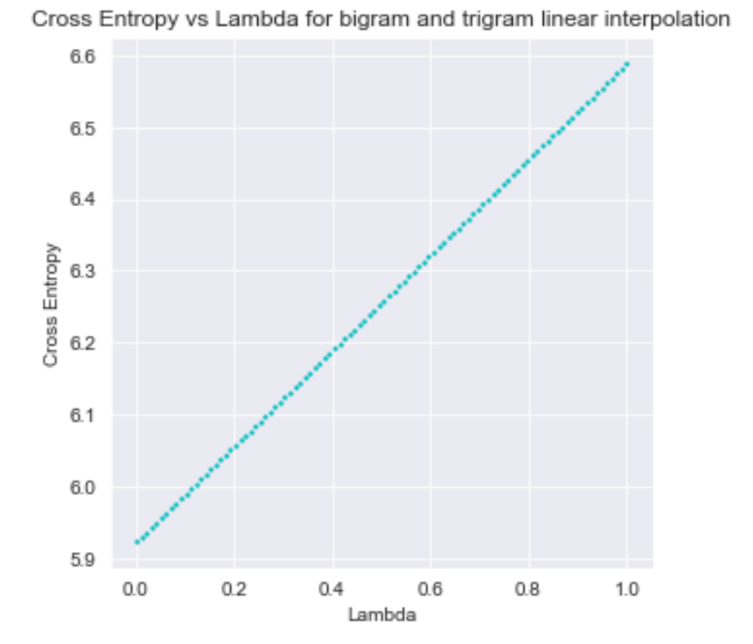
## Mmm

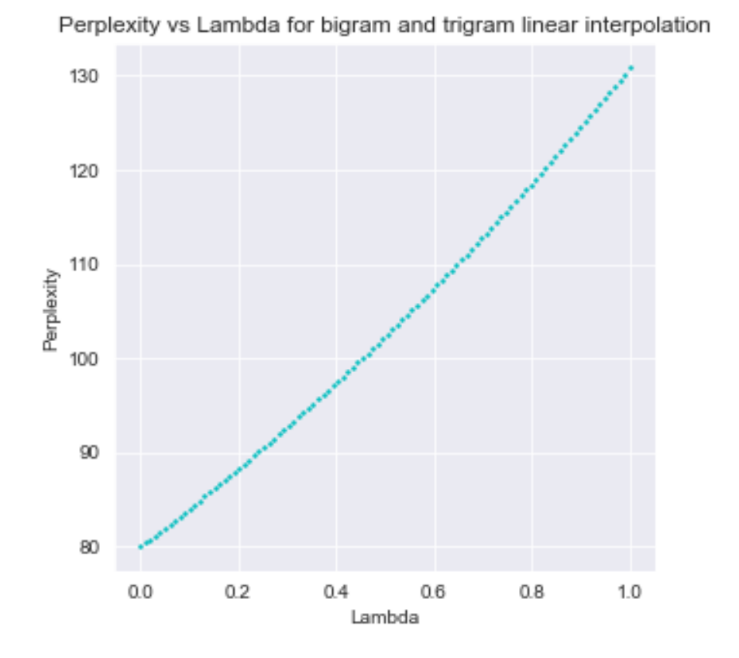
## Bbb

## Aaa

## Linear Interpolation

For the final question, the Bigram and Trigram models were combined using Linear Interpolation. The notation was applied for 100 values of the hyperparameter λ between 0.001 and 1. For values of the hyperparameter close to 0 the Bigram model is favored, while for values close to 1 the Trigram model is favored.

For the probability calculations of the two models, the bigram\_prob() and the trigram\_prob() methods were used. The value of the parameter alpha of Laplace Smoothing for each of the methods was set equal to the value that was found best during the training phase. Tuning of λ was performed based on the Development Set. Finally, the different values of λ were plotted against the corresponding values of Perplexity and Cross entropy, giving the diagrams below.

As it can be observed, as λ gets higher values the perplexity and cross entropy get higher values too, meaning that the best interpolated model is the one that has a zero λ value. This is to be expected, since it was found that Bigram model is better than Trigram. By eliminating the latter form the interpolation, the model becomes the same as the Bigram model and thus, better.