CMPUT 501 - Assignment 3
Ngram Language Models Report
Mohammad Karimiabdolmaleki & Ali Zamani

- How did you select the final sets of parameters (N)? What are the values?

To find the optimal N, we implemented a function that loops through N values for each smoothing method function and compares the obtained accuracy to find the maximum accuracy on the development set. For the unsmoothed model, N=1 resulted in an accuracy of 91%, for the Laplace smoothing model, N=4 resulted in an accuracy of 98%, and for the interpolation smoothing, we achieved an accuracy of 94% with N=3.

- Which model performed the best? Discuss the relative performance of the smoothing variants and n-gram settings.

- Across different smoothing methods, Laplace (add-one) smoothing model performance with N=4 was superior. The model's performance decreased with lower N values due to the limited context and decreased with more significant Ns due to the occurrence of unknown ngrams.

- The interpolation smoothing was placed after Laplace based on its performance on development files. N=3 was selected as the optimal N for this method. Again, the performance of the interpolated language model decreased with values lower and greater than N=3. The corresponding formula of the interpolation smoothing with N=3 as given in the textbook is equal to:

$$P(w_n|w_{n-2}w_{n-1}) = \lambda_1 P(w_n|w_{n-2}w_{n-1}) + \lambda_2 P(w_n|w_{n-1}) + \lambda_3 P(w_n)$$

- It seems reasonable that the performance of the interpolation smoothing should decrease with lower N since it will lose the functionality of the trigrams. However, by increasing the N to 4, the performance also decreases since it turns out that there would be some fourgrams that are not present in the limited amount of text of the training corpus. This would result in having a 0 probability times other probabilities, which results in 0. The reason that Laplace performed better was that it incremented all of the zero possibilities by 1.

- Although the performance of the unsmoothed model is acceptable, it had the lowest accuracy among all of the models, and its best performance was achieved using N=1. One of the main reasons this model performed worse is that there were sequences in the development files that did not exist in the training file, which resulted in a zero probability. Even with increasing the N, the result becomes worse since we are forming different combinations not present in the training data.