Assignment 2 Report

# Approach

To calculate the probabilities of the unigram model I have converted the sentences of the training set into individual words and then converted each word into a character, count the number of times that each character occurs and then calculate the probability.

P( ci ) = count ( ci ) ) / count ( total number of characters )

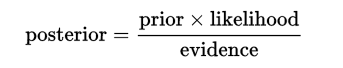
Similarly, for bigram model I have have converted the sentences of the training set into individual words and then converted each word into a character, count the number of times that each character occurs and then calculate the probability.

P( ci | ci-1 ) = count ( ci-1, ci ) / count ( ci-1 )

Similarly, for trigram model I have have converted the sentences of the training set into individual words and then converted each word into a character, count the number of times that each character occurs and then calculate the probability.

P( ci | ci-1 ci-2 ) = count (ci-2,ci-1, ci ) / count (ci-2 ci-1 )

In plain English it can be written as:



# Challenges

* The main challenge of the assignment was to covert a set of sentences into a single character and then calculate the probability. To overcome this, I had to understand how to implement FreqDist and ConditionalFreqDist into the program. FreqDist will be used to compute number of occurrences of each item in the list (like in the case of unigram) whereas ConditionalFreqDist is a collection of frequency distribution, each one for different condition (like in the case of bigram and trigram).
* The other challenge was calculating the accuracy of the models based on the predictions. To overcome this, I used the formula mentioned by Naveen, Accuracy= (Correct Prediction \* 100)/Size of the Data Set

And multiplied it by 100 to get the accuracy of the model in terms of percentage.

* Apart from this there was the challenge to develop the bigram, trigram character model in the first place to calculate the probability. NLTK book explains it in Character 1 and Character 2.

# Model Comparison

|  |  |  |  |
| --- | --- | --- | --- |
| Language Models | Unigram | Bigram | Trigram |
| English Vs English | 95% | 80.6% | 62.7% |
| English Vs French | 77.9% | 63.8% | 46.9% |
| Spanish Vs Spanish | 96.7% | 80.2% | 64% |
| Spanish Vs Italian | 92.3% | 51.6% | 36% |

From the above table we can conclude that is harder to distinguish between Spanish Vs Italian as compared to English Vs French in a Unigram Model. But for bigram and trigram models English Vs French is better than Spanish Vs Italian. Spanish Vs Italian words are different but the characters are very similar so the unigram accuracy is high and since the characters are same and not the words, there is a dip in terms of percentage of bigram and further in trigram when compared to English Vs French. But in English vs French model words such as justice, aspiration, nations, social, organization etc. are similar in English Vs French(have the same meaning) but then there are words such as plus, dont etc. (have different meaning) present in the both languages so the bigram and trigram accuracy is higher.