ASSIGNMENT – 1

- EAVANSHI ARORA
- 201501115

1) **Tokenisation** : The data contains many special characters and links. First we split the lines on (\n | \t). Whenever we encounter any special character, we replace it with a null character. And for all the other words we separate them on space and add them to the dictionary.

Some problems faced during tokenisation are:

a.) Handling Abbreviations - When a period follows an abbreviation it is an integral part of this abbreviation and should be tokenized together with it unlike end of a line in which both the words are separated.

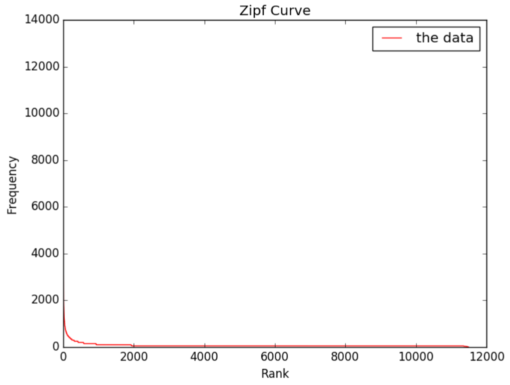
b.) Handling Hyphenated Words – There are many types of hyphens:

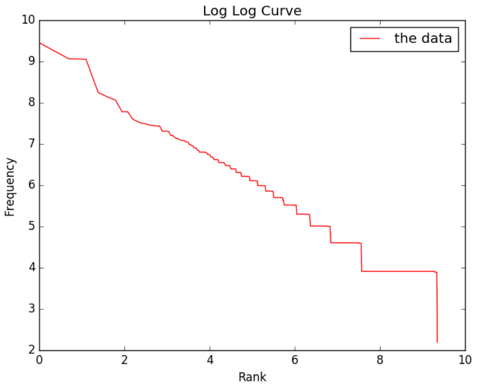
- End-of-Line Hyphen - End-of-line hyphens are used for splitting whole words into parts to perform justification of text during typesetting and should be removed during tokenisation
- True Hyphen – Integral part of words and should not be removed during tokenisation.
- Lexical Hyphen - certain prefixes (and less commonly suffixes) are often written hyphenated, e.g. co-, pre-, meta-, multi-, etc.

c.) Numerical and special expressions like Email addresses , URLs , Complex enumeration of items , Telephone Numbers , Dates , Time ,Measures ,Vehicle Licence Numbers ,Paper and book citations ,etc. These can produce a lot of confusion to a tokenizer.

2) **Creating Unigram , Bigram and Trigram :** For unigrams, we separate each word on space and add its count to the dictionary. For bigram we maintain a previous word for each current word and similarly insert it in the dictionary as a tuple of two words. Similarly for trigrams we maintain two previous words.

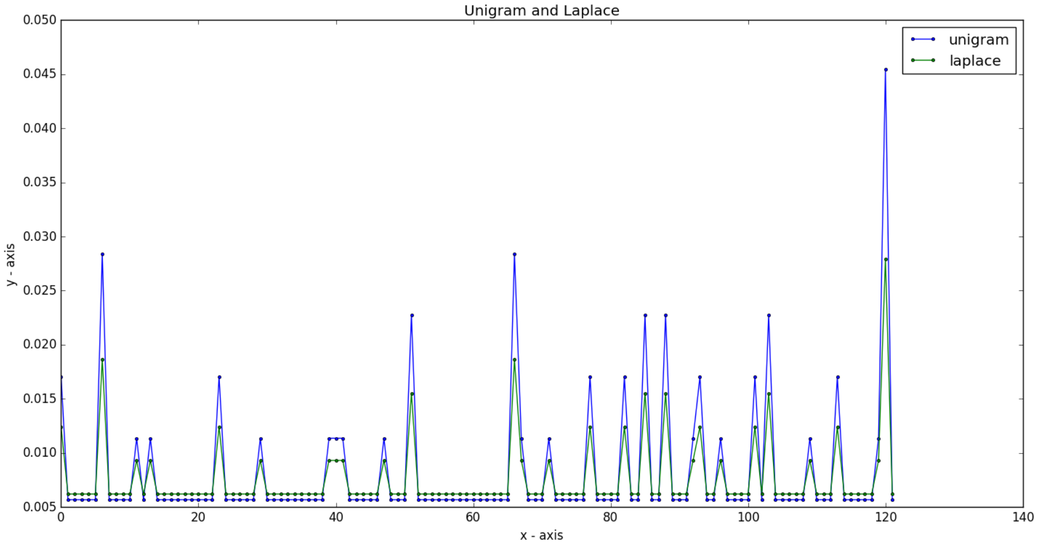
For **Unigram**, we sort the the words according to their count and give them ranks (Rank 1 to most occuring element). Then we plot a graph with y-axis as frequency and x-axis as rank of words. We obtain the following graph for Unigram on dataset given (left) and on taking log of both rank and frequency we obtain an almost straight line graphs as follows (right):



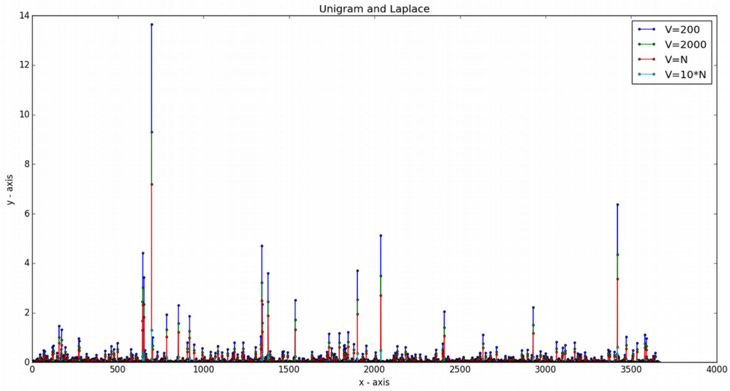


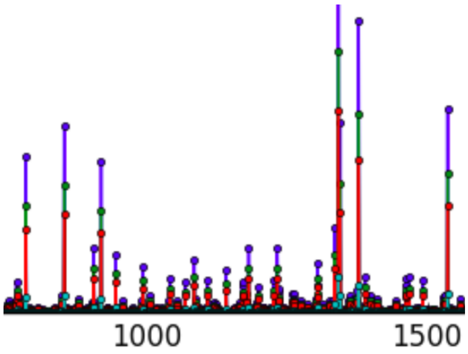
We obtain the same graph for **Bigram** and **Trigram** as well. Zipf's law states that given some corpus of natural language utterances, the frequency of any word is inversely proportional to its rank in the frequency table. Thus the most frequent word will occur approximately twice as often as the second most frequent word, three times as often as the third most frequent word, etc.

3.) **Laplace Smoothing :** Laplace smoothing smoothing proceeds from a logic of slightly correcting the observed proportions (in the case of categorical variables) in the direction of a uniform distribution among the categories (i.e., injecting a bit of equi-probability among them) . Here is a graph showing the difference b/w the original and the smoothed values of a **unigram**. We observe that the smoothed values of big values decreases slightly whereas that of small values increases slightly.

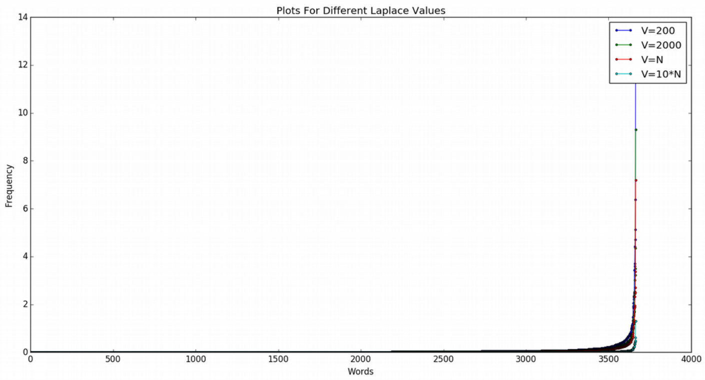


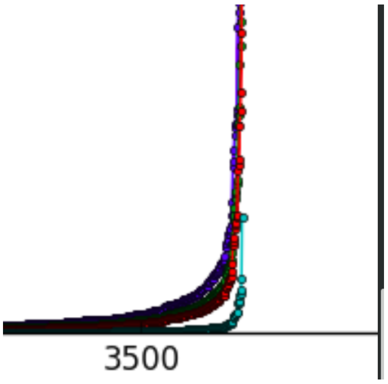
Now for different values of V, when we plot the graph we get as follows (for **Unigram**):



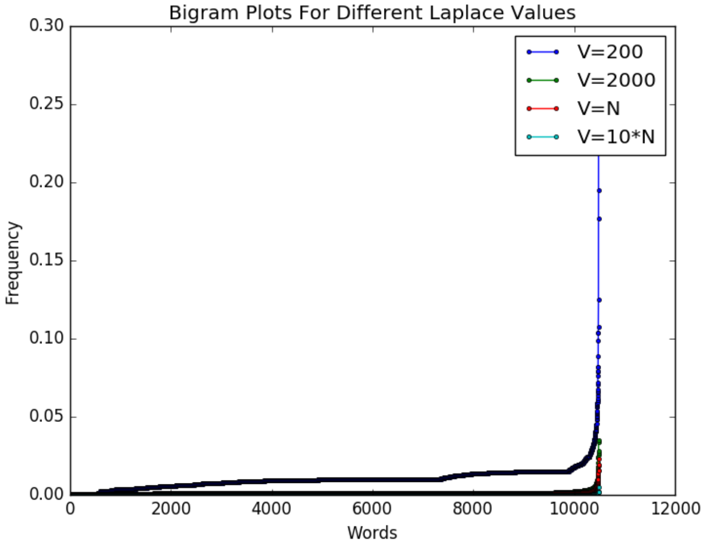


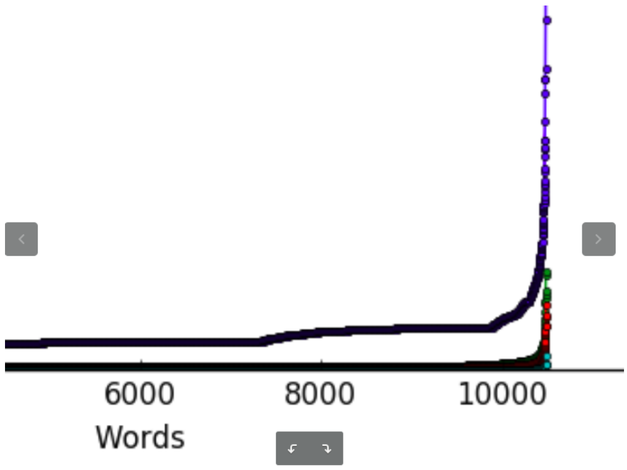
When we sort the values, we get a graph like this:



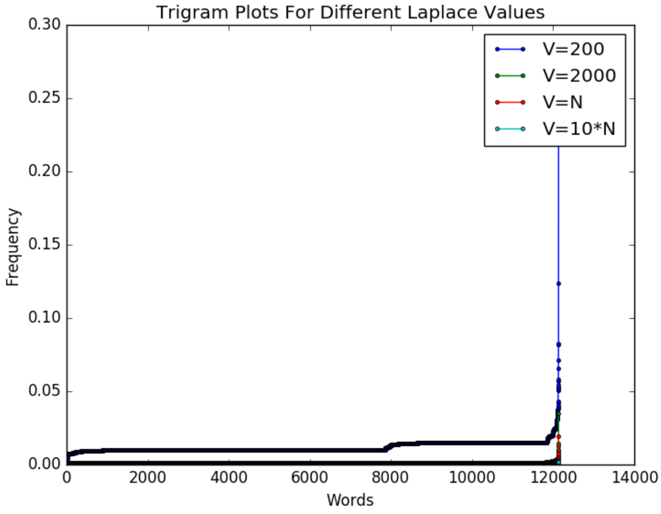


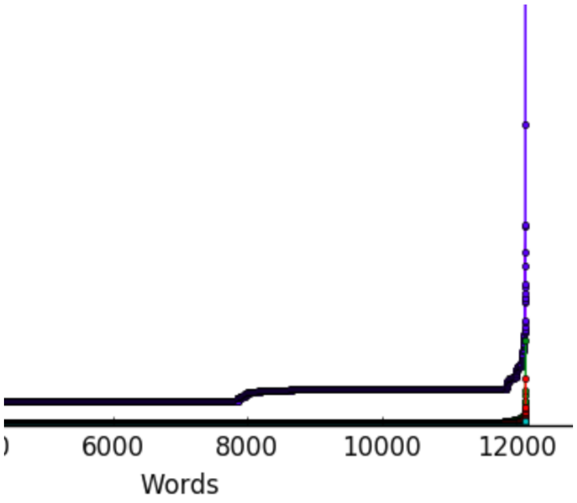
For **Bigram**:



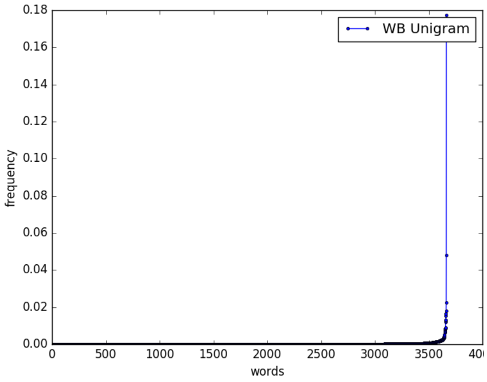


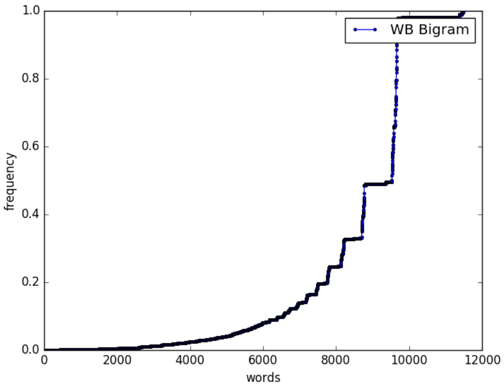
For **Trigram:**

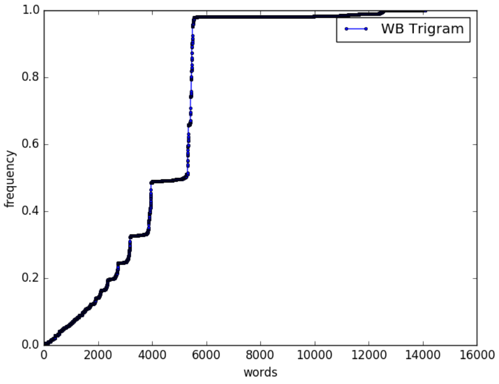




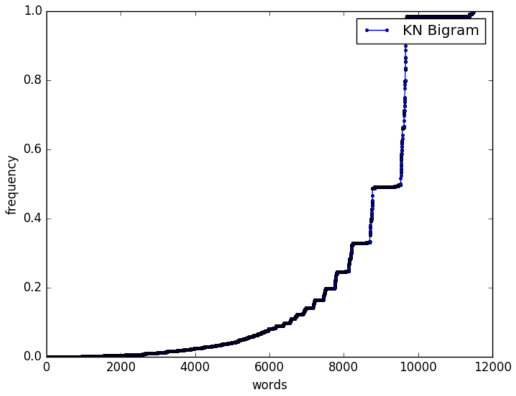
4.) **Witten Bell Smoothing :** ”Witten-Bell”-smoothing is an instance of the recursive interpolation method. Probability mass is shifted depending on context of words.

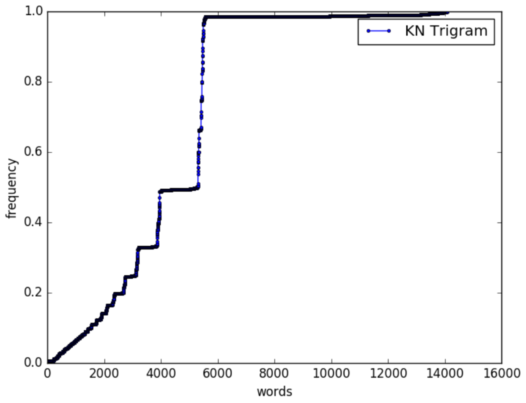




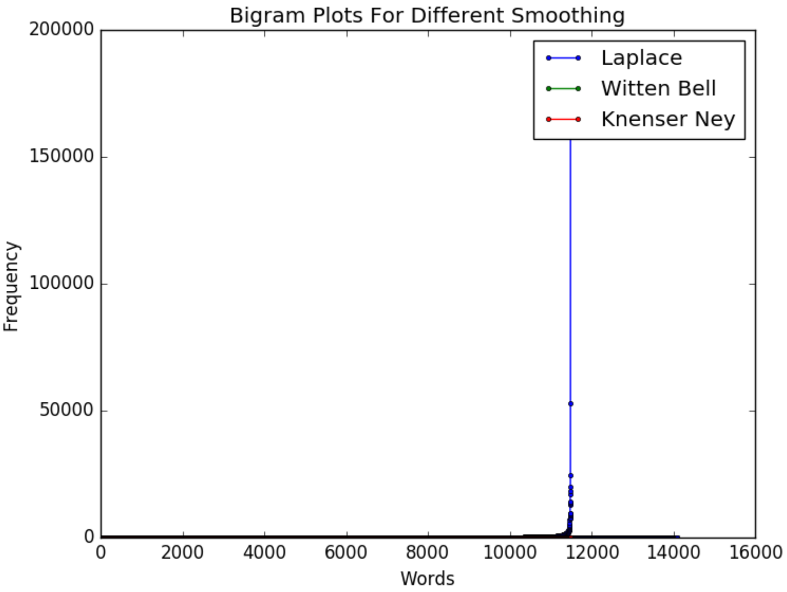


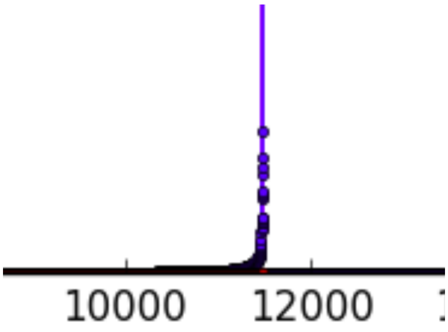
5.) Kneser Ney Smoothing :





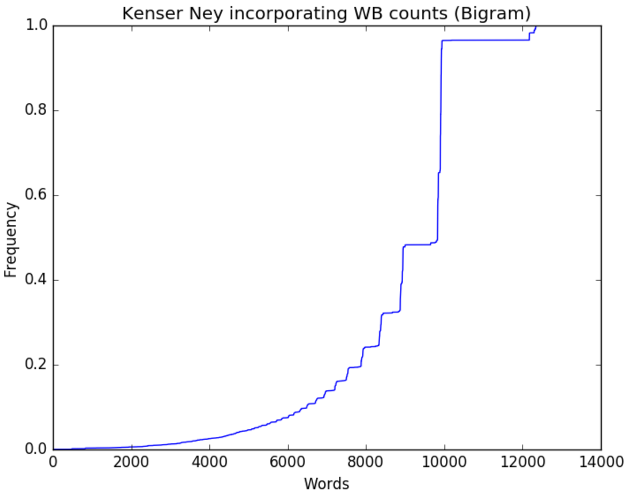
6.) Comparisons between the three smoothings:

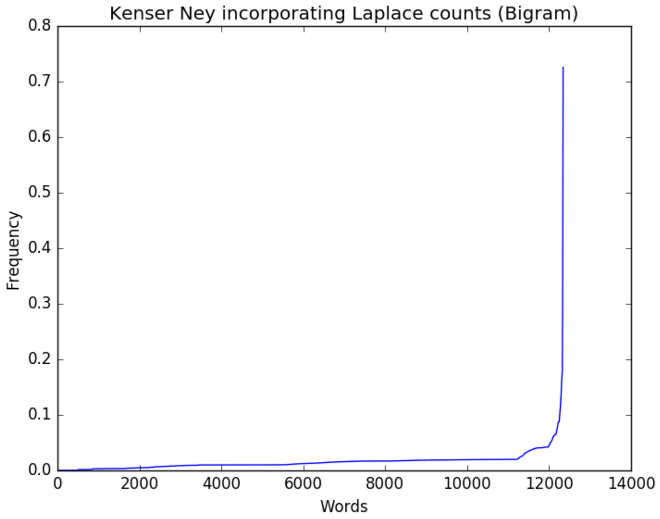




7.) Using the estimates from laplace and wittenbell in the absolute discounting step :

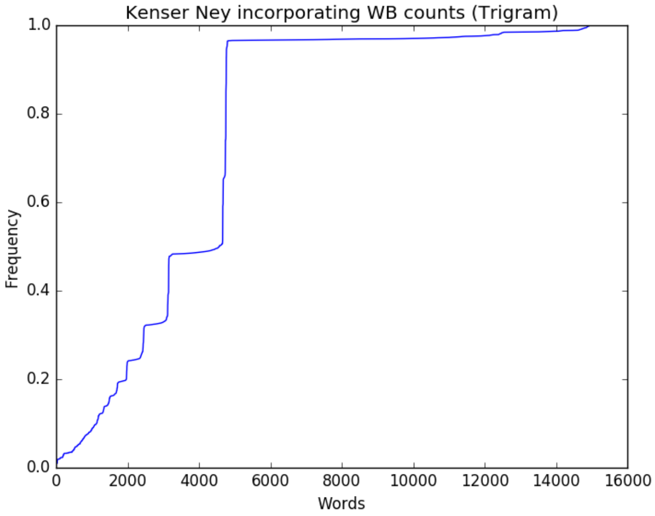
For Bigram:

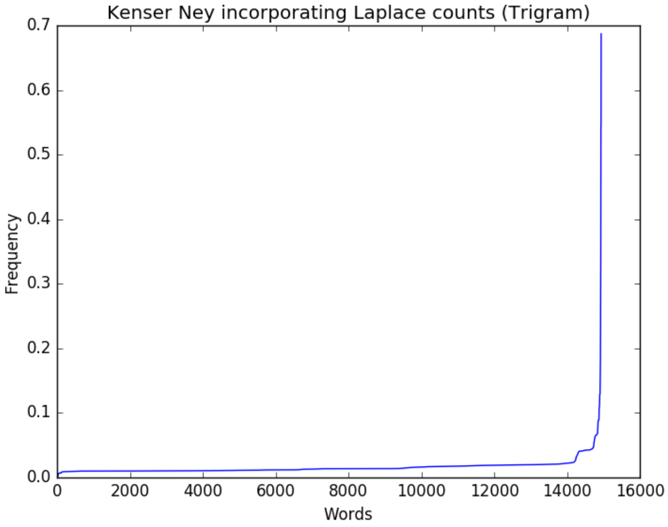




For

Trigram:





8.) Kneser Ney Prediction Trigram:

News: i m not a thing
Movies: i m not the best to ever do it
Anime: i m not sure i have n t think i ve been meaning to watch this for a while now i ll probably be too busy to watch anime i m looking for a good one punch man

Kneser Ney Prediction Bigram:

News: i m not a lot of the fuck
Movies: i m not the first time
Anime: i imgur com watch v xf2ynhaeqvk