**Restaurant Reviews: NLP**

**About the dataset:** The dataset consists of 1000 reviews and for each of the review we have a result 0/1 i.e. whether the review is positive or not. The dataset was in a .csv file, which was converted into a .tsv file since a review can have a ‘comma’ that would have eventually messed up the whole model. Also, I ignored all the texts within the quotes to make the model simpler.

**Cleaning the dataset:** To clean the text, I imported the library ‘re’. To make the most out of the dataset, following steps were taken for cleaning.

1. Removed the numbers and punctuation marks from all the reviews and kept only the alphabets.
2. Converted all the texts into lower case.
3. Tokenization
4. Stemming and removing the stop words.
5. Joining back different tokens into the review, but now without the stop words and in root form.

**Creating a bag of words model:** After cleaning the dataset, I created a corpus containing 1000 cleaned reviews. So, it is from the corpus that I will create a bag of words model. In a bag of words, all the unique words of 1000 reviews will be taken and create a column for each of the word whereas the rows will be nothing but the 1000 reviews. Therefore, each cell in a table will correspond to a specific review and a specific word. In each cell, there will be a number which will tell the number of times that particular word appears in that review. This will create a sparse matrix and that is why stemming was necessary, so that we can reduce the sparsity of the matrix. To keep the matrix limited and reduce sparsity, I am using only 1500 most frequent words within the review.

**Results:**

Test size is taken as 15% since there were only 1000 reviews, which are very less to give correct predictions. Based on research, I am only going to use Naïve Bayes and Random Forest models to predict results.

1. **Using Naïve Bayes:**

**A screenshot of a cell phone

Description automatically generated**

**Accuracy =** (38+70)/150

**= 0.72**

1. **Using Random Forest:**

**A screenshot of a cell phone

Description automatically generated**

**Accuracy =** (60+44)/150

**=0.69**