**What is NLP?**

According to \_, Natural Language Processing (NLP) is \_. Practical applications of NLP are Google Voice search, sentiment analysis, Siri, etc.  
NLP are used to extract meaningful data from textual data.

NLP pipeline:   
What is a pipeline? A pipeline is just a way to design a program/system, where the output of a module/system is feed into another module/system.

Raw Text Document -> Sentence Segmentation -> Tokenization -> Part-of-speech Tagging -> Lemmatization -> Stop Words -> Dependency parsing -> Noun Phrases -> Name Identity Recognition -> Processes Text

[Google each module in the pipeline and write a small description of each and the benefits]

Data cleaning

In data cleaning, we convert the raw text into a list of words that are free of unnecessary information that might skew the final results.

Example of data cleaning techniques of data cleaning are Tokenization, Stop words removal, Stemming etc.

**What is NLTK?**

The Natural Language Processing Tool Kit (NLTK) is a platform used for building Python programs that work with human language data for applying statistical natural language processing (NLP).

**Naïve Bayes Algorithm**

Naïve Bayes classifiers are a family of simple “probabilistic classifiers” based on applying Bayes Theorem with strong independence assumptions between the features.

Classifiers in simple terms group data with similar characteristics together.

Bayes Theorem is the principle that drives the Naïve Bayes classifier. Bayes Theorem is used to calculate the conditional probability.

Bayes Theorem:

Which in simple English term can be read as: Given events A and B, the probability that event A will happen given that event B has already happen is equal to the probability of event B happening given that event A has already happened multiply by probability of event A happening, all over (divided by) probability of even B happening. “|” is known as the conditional probability.

So how is the Bayes Theorem used for classification? Good Question:

Imagine a spam filtering systems, with x is the text to be classified (i.e. x = {text data}) and y is the classification of the text (i.e. y= {1, 0}, where y = 1 for a text labelled as spam and y = 0 for a text labelled as not spam).

Therefore, the probability of x being a spam (y = 1) given text x, is equal to the probability of text x given that text x is a spam multiply by the probability of spams (y = 1) all over probability of x.

And the probability of x not being a spam (y = 0) given text x, is equal to the probability of text x given that text x is not a spam multiply by the probability of not spams (y = 0) all over probability of x.

To classify the text x, we take the probability (i.e. the probability between the probability of x being a spam (y = 1) given text x and the probability of x not being a spam (y = 0) given text x) with the highest result as facts. The probability is called “Likelihood”, and this is gotten from the data this model trains on.

Probability of spam is also found from the training data, where:

And

And we calculate this by:

And,

Where n is the number of text x, and y(i) is the associated classification that comes with the text x, and x(i) is the text on line number “i” in this particular context.

To find probability of text x given that text x is a spam, we do, given the word x(i) in the set of words x:

Which can be interpreted as probability of text x is spam given that x is already labelled as spam is equal to probability that words within the text x are spam given that the other words within the text x are spam words as well, but we avoid doing this because, given |x| is huge, it becomes computationally expensive to find the probability if every word in set x is dependent on other words within set x.

So instead, we use a Naïve Bayes assumption, which is, the probability of a word being spam is independent of another word with the same set being spam:

This can be mathematically for probability of text x is spam given that x is already labelled as spam explained by:

And, probability of text x is not spam given that x is already labelled as not spam,

**Sentiment Analysis**

[Google sentiment analysis and give a description]