Part B

Naive Bayes is a basic but effective probabilistic classification model in machine learning that draws influence from Bayes Theorem.

$$P(y|X) = \frac{P(X|y).P(X)}{P(y)}$$

Features

- The Naive Bayes method makes the assumption that the predictors contribute equally and independently to selecting the output class.

a) Gaussian Naïve Bayes:

When working with continuous data, one common assumption is that the continuous values associated with each class follow a normal (or Gaussian) distribution.

b) Multinomial Naïve Bayes:

This classifier makes use of a multinomial distribution and is often used to solve issues involving document or text classification. For discrete data like word counts, multinomial classification works best. It disregards the absence of the features. As a result, if the frequency is 0, the likelihood of that feature occurring is also 0. Multinomial naive Bayes ignores that feature.

c) Complement Naïve Bayes:

This method is nearly identical to the Multinomial, with the exception that we now count the occurrences of a word in the complement to the class. With unstable data, Multinomial Naive Bayes does not perform well.

d) Bernoulli Naïve Bayes:

When the predictors are boolean in nature and are supposed to follow the Bernoulli distribution, this classifier is utilized. When characteristics are binary, the Bernoulli formula is similar to the multinomial one. If you have discrete features in 1s and 0s that signify the existence or absence of a feature, you can use them instead of the frequency of the word. In that instance, the features will be binary, and Bernoulli Naive Bayes will be used.

e) Categorical Naïve Bayes:

The categorical Naive Bayes classifier is well suited to classification with categorically distributed discrete features. Each feature's categories are chosen from a categorical distribution.

The probability of category t in feature i given class c is estimated as:

$$P(x_i = t \mid y = c \,;\, \alpha) = \frac{N_{tic} + \alpha}{N_c + \alpha n_i},$$

f) Out-of-core Naïve Bayes model fitting:

Various nave Bayes methods can be used to solve large-scale classification issues where the entire training data is too vast to fit in memory. As a result, most types supply a partial fit technique for this problem, which may be used progressively like other classifiers, as seen in Out-of-core classification of text documents.

Jaccard Similarity is given by:

$$s_{ij} = \frac{p}{p+q+r}$$

p = # of attributes positive for both objects
q = # of attributes 1 for i and 0 for j
r = # of attributes 0 for i and 1 for j

The **cosine similarity** is measure the cosine angle between the two vectors. For cosien we have to convert all sentences to vectors. For converting to vector we can use TF-IDF, Word2Vec. cosine similarity is:

$$\frac{A \cdot B}{\|A\| \|B\|}$$

where A and B are object vectors.

Major difference between jaccard and cosine similarity:-

1. Jaccard Similarity takes set of unique length of words instead cosine similarity takes whole sentence vector
2. If data duplication is not matter then its better to use jaccard similarity else cosine similarity is good for measuring the similarity between two vectors even if the data duplication is there.