2. Training a Naïve Bayes Classifier using top-K frequent words

Model: ScikitLearn's Gaussian Naïve Bayes Classifier

Naïve Bayes:

$$P(y|x) = \frac{\prod_{i=1}^{n} P(x_i|y)\,P(y)}{P(x)} \text{ where } n \text{ is the number of features}$$

Gaussian distribution is assumed as the likelihood of the features:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}}\exp\left(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}\right)$$

The parameters σ_y and μ_y are estimated using maximum likelihood.

Program Flow

1. the frequency distribution of word id in each sample is calculated using nltk
2. the feature of each sample is generated using the number of times each word id appeared in the sample, while only considering top-k most frequent word id
3. the model is fitted to training dataset, then evaluated with the test dataset

Result

Figure below shows the result when k=100, 1000, 10000 respectively.

```
Calculating freqdist of x_train & x_test...done.
~~~~~~~~~~~  K = 100  ~~~~~~~~~~~
Obtaining frequency of top-100 words in x_train...done.
Obtaining frequency of top-100 words in x_test...done.
Training gnb model...done.
Accuracy = 0.69168, Precision = 0.70542, Recall = 0.65824

~~~~~~~~~~~  K = 1000  ~~~~~~~~~~~
Obtaining frequency of top-1000 words in x_train...done.
Obtaining frequency of top-1000 words in x_test...done.
Training gnb model...done.
Accuracy = 0.81004, Precision = 0.82396, Recall = 0.78856

~~~~~~~~~~~  K = 10000  ~~~~~~~~~~~
Obtaining frequency of top-10000 words in x_train...done.
Obtaining frequency of top-10000 words in x_test...done.
Training gnb model...done.
Accuracy = 0.66128, Precision = 0.76809, Recall = 0.46208

Press any key to exit.
```

The performance of the model decreased when k is increased from 1000 to 10000.

It is likely due to overfitting.