Documentation

Introduction:

Text categorization is the task of deciding whether a piece of text belongs to any of a set of prespecified categories. It is a generic text processing task useful in indexing documents for later retrieval, as a stage in natural language processing systems, for content analysis, and in many other roles.

The use of standard, widely distributed test collections has been a considerable aid in the development of algorithms for the related task of text retrieval (finding documents that satisfy a particular user's information need, usually expressed in an textual request). Text retrieval test collections have allowed the comparison of algorithms developed by a variety of researchers around the world.

The Reuters-21578 collection is distributed in 22 files. Each of the first 21 files (reut2-000.sgm through reut2-020.sgm) contain 1000 documents, while the last (reut2-021.sgm) contains 578 documents.

Splits:

There are several splits to the data. Apart from the normal Train-test_split, which divides the data randomly in the ratio 3:1, there are Lewis Split(Modified Apte Split) and CGI Split.

Packages used:

pandas for managing dataframes
re for regex
sklearn for machine learning
sklearn.pipeline for Pipeline
sklearn.feature_extraction.text for TfidfTransformer,TfidfVectorizer
sklearn.metrics for Accuracy
sklearn.naive_bayes for MultinomialNB and ComplementNB
skmultilearn.problem_transform for LabelPowerset
sklearn.preprocessing for MultiLabelBinarizer

Algorithm:

Here we use the Naive Bayes Classifier to classify the documents into the given categories. A document may belong to more than one category.

Among the various Naive Bayes Classifiers available, we use the Multinomial Naive Bayes and Complement Naive Bayes models. Complement Naive Bayes works better in case of skewed datasets.

We remove the repititions and the rows which have no topics. The topics have been encoded using label encoder and the documents have been vectorised using Tf-Idf vectoriser.

The Multinomial Naive Bayes and Complement Naive Bayes models are used in each of the three splits. Among the models Complement Naive Bayes gives the best results when we use Lewis Split.

However the above models only give us single classes as outputs.

For multi-label classification we use the multilabel binarizer to tranform the topic to a binaru sparse array. Then using the Complement Naive Bayes through the labelpowerset classifier we find multi-label claffifications.

Among the Multi-Label classifiers the Complement Naive Bayes gives the highest accuracy for Train-Test Split(0.25) and Lewis Split.

Results:

The accuracies of the models using various splits are as follows:

Split \ Model	**Multinomial Naive Bayes**	**Complement Naive Bayes**
Train-Test Split (0.25)	64.57%	81.91%
CGI Split	32.30%	64.59%
Lewis Split	64.30%	81.92%

Split \ Multi-label Model	**Multinomial Naive Bayes**	**Complement Naive Bayes**
Train-Test Split (0.25)	64.46%	83.80%
CGI Split	33.21%	66.42%
Lewis Split	65.50%	83.35%