

Structure of Code:

- The code is designed using Python and contains a function for data handling
- First, the input folders are retrieved as a list from the parent directory.
- Then the model starts with training of first 500 files in each category where the words and its frequency are stored in a dictionary for easy access to test it in the future
- At the end of training model, we will have dictionary of trained files under each category, dictionary of words under each category and a master dictionary of total words.
- Now, using the trained model we test the remaining 500 files in each category and classify the words by computing log probabilities(because it improves accuracy of model) over the testing data using the formula,

$$P(i | j) = \frac{\text{word}_{ij} + \alpha}{\text{word}_j + |N| + 1}$$

Where N is the total number of words in vocabulary and $\alpha = 0.0001$ (Laplace smoothing)

$$P(j) = \log \pi_j + \sum_{n=1}^N \log(1 + f_i) \log(P(i|j))$$

Which is the optimal model.

- The number of collisions of maximum probabilities of a particular word in a category is computed and is divided by the total population in order to calculate accuracy of the model.
- Accuracy of this model comes out be **65.00 %**

Screenshot of output:

The screenshot displays the Spyder Python IDE interface. The main editor shows the `Naive.py` script, which implements a Naive Bayes classifier for text classification. The script includes imports for `os`, `copy`, and `math`. It defines a `data_handler` function to process text by removing special characters and stop words. The main logic iterates through folders, training on the first 500 files and testing on the remaining 500 files. The output in the IPython console shows the training and testing process, including the total number of words found (104163) and the final accuracy of 65.0%.

Variable explorer:

Name	Type	Size	Value
accuracy	float	1	65.0
alpha	float	1	0.0001
category	int	1	20
count	int	1	500
dic_of_dict	dict	20	{'alt.atheism':{...
file	str	1	84569
file_data	str	1	xref cnloe rv c...
file_path	str	1	20_newsgroups/ta...

IPython console:

```
In [7]: runfile('C:/Users/Balaji/Downloads/ML/P2/Naive.py', wdir='C:/Users/Balaji/Downloads/ML/P2')
Training the first 500 files
Processing the text.....
Total number of words found in the dataset : 104163
Testing the remaining 500 files
Calculating probabilities.....
Accuracy : 65.0%

In [8]:
```

Permissions: RW End-of-lines: CRLF Encoding: ASCII Line: 42 Column: 26 Memory: 70 %