Machine Learning
Project-2 News Text Classification
Using Naïve Bayes

Saachi Shivhare - 1001830083

Abstract

This report summarizes the analysis done for Text Classification using Naïve Bayes Classifier. Naïve Bayes is done on given set of training data and probability is calculated to predict the class values of the unseen test data. Accuracy is also computed to check the efficiency of the classifier.

1. Introduction

With the explosive growth of online information, it is not easy for people to figure out which information is in his/her interests. For instance, thousands of articles in newsgroups often let one very hard to pick up something he/she really would like to see. Thus, it will be nice if we have tools which can group huge number of documents into different categories or classes. This process is known as Text categorization. The problem in this is to classify documents into fixed number of classes. Naïve Bayes Algorithm can be used to perform text classification.

Naive Bayes computes probabilities of the word to estimate the likelihood that a given document belongs to a particular class.
This probability estimate is used for decision making. We have applied same approach to train the classifier.

2. Dataset

The dataset being used in the project is 20 Newsgroup Dataset. The dataset consists of 20 different newsgroups, each consisting of 1000 messages. The dataset can be found at http://www.cs.cmu.edu/afs/cs/project/theo-11/www/naivebayes.html

location. In this project, I have first separated the training data and the test data. The training data can be found at “train_data” folder and the testing data can be found at “20_newsgroups” folder.

3. Naïve Bayes Classifier

A classifier is a machine learning model that is used to discriminate different objects based on certain features. The crux of the naïve bayes classifier is based on the Bayes theorem.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Using Bayes theorem, we can find the probability of A happening, given that B has occurred. Here, B is the evidence and A is the hypothesis. The assumption made here is that the predictors/features are independent. That is presence of one feature does not affect the other. Hence it is called naive.

4. Preprocessing

To develop a fair classifier, I have used random function from random library that data is divided randomly between two sets.

In a text document, we have a lot of words for example, “the”, “was”, “could”, “me” etc. which are not useful when classifying a document. Below is the list of all such words and punctuation marks that are not included while preparing feature list.

'i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', 'she', "she's", 'her', 'hers', 'herself', 'it', "it's", 'its', 'itself', 'they', 'them', 'their', 'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll", 'these', 'those', 'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had', 'having', 'do', 'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because', 'as', 'until', 'while', 'of', 'at', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through', 'during', 'before', 'after', 'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over', 'under', 'again', 'further', 'then', 'once', 'here', 'there', 'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each', 'few', 'more', 'most', 'other', 'some', 'such', 'no', 'nor', 'not', 'only', 'own', 'same', 'so', 'than', 'too', 'very', 's', 't', 'can', 'will', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'll', 'm', 'o', 're', 've', 'y', 'ain', 'aren', "aren't", 'couldn', "couldn't", 'didn', "didn't", 'doesn', "doesn't", 'hadn', "hadn't", 'hasn', "hasn't", 'haven', "haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't", 'mustn', "mustn't", 'needn', "needn't", 'shan', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't", 'weren', "weren't", 'won', "won't", 'wouldn', "wouldn't", '!', '"', '#', '$', '%', '&', "'", '(', ')', '*', '+', ',', '-', '.', '/', ':', ';', '<', '=', '>', '?', '@', '[', '\\', ']', '^', '_', '`', '{', '|', '}', '~'

Also, we need to change the case of every word as it might happen that a word is mentioned in both the cases. If we do not change the case to lower case while counting the number of words, we will end up considering same word as two features.
I also used strip function to remove any special characters from the word. If the word is present in the feature list, increase the count and if it is not present then we add the new word.

I have also removed the less frequent words from the feature list.

5. Training the Classifier

After preprocessing, I trained the model. In this I computed the count of a particular word in a particular category and also compute the probability of that particular word in that particular category. Training of the classifier is completed once the probabilities of all the words in all the categories are calculated.

6. Testing the Classifier

While testing the model, I calculated the logarithmic value of the probability for all the words of all the categories present in "20_newsgroups" location. Results are stored in a dictionary. If the category matches the category of the data, count of correctly classified files is incremented and at the same time decreasing the overall error of the classifier. Accuracy is calculated using below formula.
The average of all the categories accuracies is considered as the accuracy of the classifier.

The accuracy of the model is 80.012%

7. References

https://www.geeksforgeeks.org/os-walk-python/
https://www.geeksforgeeks.org/removing-stop-words-nltk-python/
https://www.geeksforgeeks.org/python-string-strip-2/
https://www.digitalvidya.com/blog/document-classification-python-machine-learning/
http://www.cs.columbia.edu/~evs/ml/OthelloStudProj/huang/write-up.html
https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c
https://machinelearningmastery.com/naive-bayes-classifier-scratch-python/
https://geoffruddock.com/naive-bayes-from-scratch-with-numpy/
https://realpython.com/working-with-files-in-python/
https://www.geeksforgeeks.org/python-os-path-join-method/

https://www.geeksforgeeks.org/with-statement-in-python/
https://www.tutorialspoint.com/How-to-move-a-file-from-one-folder-to-another-using-Python
https://www.youtube.com/watch?v=60pqgfT5tZM&t=428s
https://stackoverflow.com/questions/19609991/typeerror-can-only-concatenate-tuple-not-int-in-python