Project Report

Soft Skills Author Prediction

Using Naive Bayes

Submitted to :

Sir Praveen Garimella

Submitted by :

A. Sai Chaitanya (IH201685006)

Nishu Sharma (IH201685066)

Sameer Chandra Pitta (IH201685084)

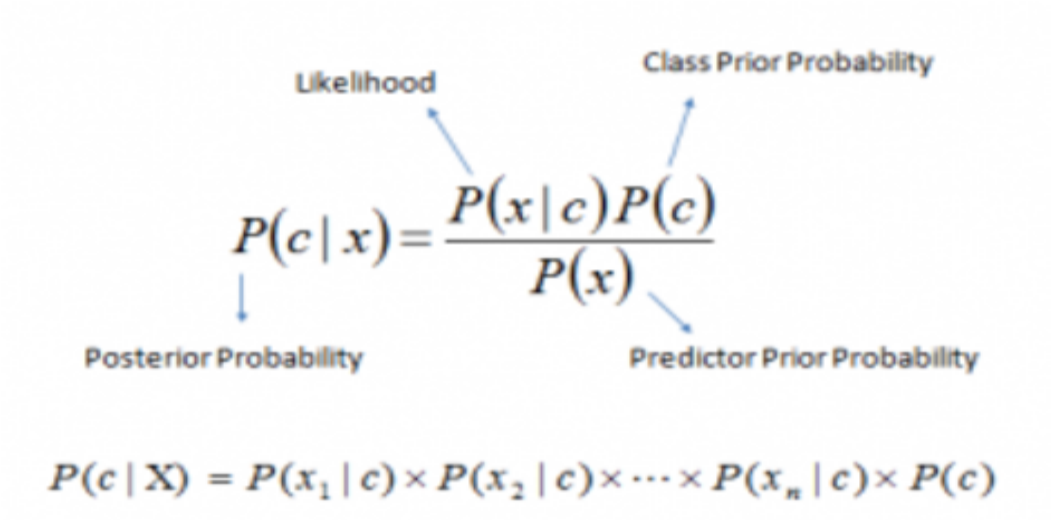
Introduction

Naive Bayes

It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c). Look at the equation below:



Above,

- *P*(c|*x*) is the posterior probability of *class* (c, *target*) given *predictor* (x, *attributes*).
- *P*(c) is the prior probability of *class*.
- *P*(*x*|c) is the likelihood which is the probability of *predictor* given *class*.
- *P*(*x*) is the prior probability of *predictor*.

Data Cleaning

Real-world data tend to be incomplete, noisy, and inconsistent. Data cleaning (or data cleansing) routines attempt to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data.

The data we were given was divided into directories based on the exam/assignment. Each directory contains the student submissions for that assignment. We faced some challenges while trying to pre-process the data. They are:

1) Not all students submitted an assignment. So the sum of submissions for an assignment is less than the total strength of the class.
2) Not every student submitted all the assignments. Most of the students didn't submit all the assignments they are supposed to submit.
3) The format of the submissions isn't the same. Some of the documents are in .docx format, some in .odt, .txt.
4) Some students submitted zip files.

Data Pre-processing:

The data pre-processing was done on the data set **manually** and through **scripting.**

Manual Pre-processing:

- In manual processing all the zip files were uncompressed and their contents renamed to reflect the standard naming conventions. In this step the corrupted zip files were removed.
- By the end of the previous step all the assignment directories had no .zip files or folders only the documents.

Scripting:

- We then wrote a python script to walk through the directories using the os package.
- In every directory we read each document using the textract package and appended it to a dictionary with the student's name as the key. Thus, at the end of the directory traversal we will have a dictionary with a student's name as the key and all the text the student wrote as the value.
- We then converted that into a csv file using csv package.

This .csv file was used to generate the Naive Bayes model to predict the author of an essay. The process for doing that is detailed in the next section.

Process

1. Load the dataset into a pandas data frame.

2. Import CountVectorizer().It tokenizes the string(separates the string into individual words) and gives an integer ID to each token.It also counts the occurrence of each of those tokens.

3. Split the dataset into test and train datasets using train_test_split from sklearn.cross_validation

4. Transform the training data and return the matrix using CountVectorizer().

5. Use the naives bayes model to fit the training data and roll numbers.

6. Then we have to make predictions on the testing data.

7. We use metrics like accuracy score, precision score, recall score, f1 score to evaluate our model.

```
('Accuracy score: ', '0.003891050583666')
('Precision score: ', '0.01556420233466')
('Recall score: ', '0.003891050583666')
('F1 score: ', '0.006225680933856')
```

Reasons for the failure of our model

As we can see, the accuracy of our model is extremely less. The possible reasons are :

- The data is too noisy
- Data for each person is inadequate i.e. we could have expected a better accuracy if we would have had more number of assignments for each person
- All the students didn't submit all the assignments. The model seems to be biased towards people who submitted more number of assignments
- Since the assignments are specific to a topic, there are a lot of words which are common across all the documents for an assignment
- Plagiarism might have also played a significant role in the failure of our model