Authorship attribution and finding whether two books are written by same author or not

By

Balaji A R

Rajarajeswari Vaidyanathan

Introduction

- With increasing amount of data available in Internet, and as most of the writings are anonymous, authorship attribution becomes important.
- Authorship attribution is a process of identifying author of the document.
- We exploit writing style present in the document to predict the author of the document.
- Every person has their unique style of writing. For example, some authors prefer to write long sentences, some prefer a concise gist.

Problem

- The proposed problem is to infer the characteristics of the written text and detect author of the document.
- Two ways to solve the problem – Supervised learning and unsupervised learning.
- Supervised Learning – Collect document samples and these samples would be used to learn a model to detect the author of the document.
- Additionally, we want to check author of the two document instances are same or not.

Goal of the project

- Can the tools and techniques from the book be useful in identifying as to who wrote them?
- The proposed project is thus to detect the author of a given text by inferring characteristics which are specific to one author. We are using supervised learning for authorship prediction and unsupervised learning for finding whether two books were written by the same author or not.
- This paper deals with similarity between authors based on their linguistic style.

Dataset used:

- In this project, we are using publically accessible book collections from “Project Gutenberg” - Gutenberg.org.
- There are around 53,000 books. The text files use the format of plain text encoded in UTF-8 wrapped at 67-70 characters with paragraphs separated by double line break. For experimental purposes, we have limited to 100 books which is of 2500 sentences in total.

Summary of the approaches taken:

Feature Extraction.

- Lexical Features – Average number of words per sentence.
- Punctuation features – Average number of commas, semicolons, and colons per sentence.
- Bag of words Features – Represents frequencies of different words in each text.
- Syntactic features – Extract syntactic features of the text.
  - POS – Classification of each token into lexical category.

Authorship prediction using SVM

- **Idea of SVM** – It helps us to clearly draw the boundaries between two classes and we need to train this model carefully to avoid overfittng.
- Performed various experiments on the features sets discussed using SVM and combined these results to improve our success rates.
- Create confusion matrix to find various measures like error rate, accuracy, sensitivity, precision.
- Precision = tp/(tp+fp).
- Recall = tp/(tp+fn)
- Accuracy = (tp + tn) / (tp+tn+fp+fn)
  - Where tp = true positive, tn = true negative
    Fp = false positive, fn = false negative.

Experiments and Results

- To perform tests, we created pair of books then extracted three types of features and performed KMeans clustering to see whether both are classified into same cluster or not.
- We have used Linear SVC from the sklearn library to classify collected texts using the extracted features.

```
Author classification started...
Processed 31 books from 4 authors with 1620817 total words in 21.909s

31 samples in 4 classes

Accuracy on training set: 1.000

Confusion Matrix:
                 alexandre-dumas  herman-melville  jane-austen  leo-tolstoy
alexandre-dumas                1                0            0            0
herman-melville                0                4            0            0
jane-austen                    0                1            2            0
leo-tolstoy                    0                1            0            4

Result:
           alexandre-dumas  herman-melville  jane-austen  leo-tolstoy
precision              1.0         0.666667     1.000000     1.000000
recall                 1.0         1.000000     0.666667     0.800000
f1                     1.0         0.800000     0.800000     0.888889

Average f1s:
0.872222222222
```

Sentence length distribution Word length distribution Pronoun distribution Conjunction distribution	52%
Hapax legomena Dis legomena Richness Readibility	35%
Hapax legomena Dis legomena Sentence length distribution Word length distribution	61.4%
Sentence length distribution Word length distribution Pronoun distribution	84%
All features	75%

Evaluation results from Author classification:

- Number of file pairs classified correctly : 25
- Number of file pairs classified incorrectly : 10
- Accurate Percentage : 71.43

Analysis:

- Most of the texts are written by alexander-dumas and herman-Melville.
- Results from SVM gave us an accuracy of 87.22%

Analysis & Conclusion

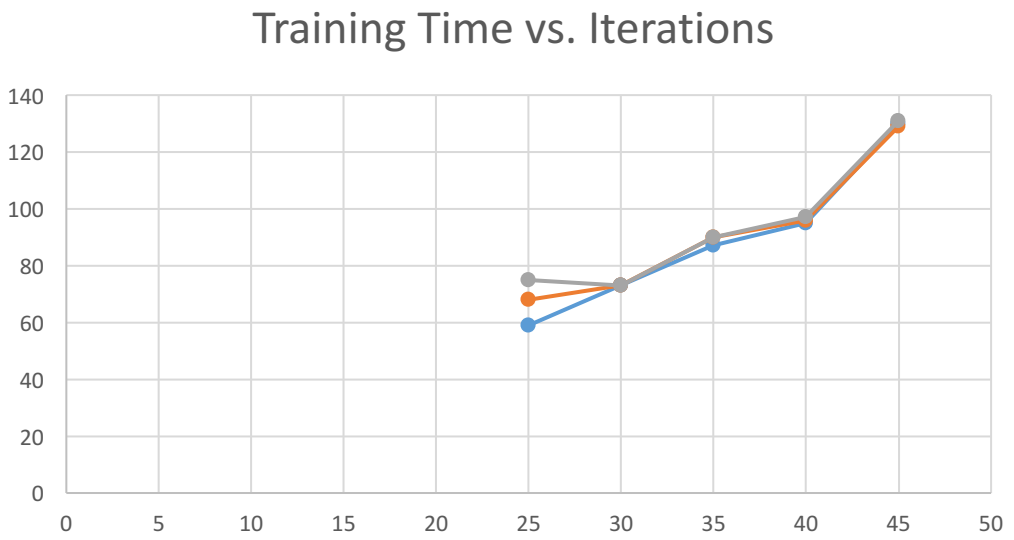
- Richness and readability helps us in identifying the author's vocabulary and it is used as a discriminating feature.
- Additionally, the function words pronoun, conjunction distribution also acts as a discriminating feature.
- According to the experiments, the success rate dramatically changes with different combinations of features.
- However the best among them are svm classifier with bag of words features, syntactic, lexical and punctuation features.
- Ran the code on AWS instance –r4.2xlarge to make use of high performance machines.

Where our solution can be used:

- Our solution assists in many other areas like plagiarism detection, email filtering, and to check the originality of the content.

Learnings and future work

- We identified that svm gives satisfactory results.
- Larger the input dataset, complexity in time increases. When handled carefully using GPU systems, we can avoid these issues.
- One possible solution to improve our results using more complex prediction models such as RNN which will again suit well for authorship attribution.



Questions?

Thank you !