HSS 322 Project Report

Authorship Identification

(AuthID)

Siddharth Chaini, 17275
Siddharth Bachoti, 17274

Github Repository : https://github.com/AKnightWing/AuthID

Acknowledgement

We thank Dr. Rajakrishnan Rajkumar for the course on computational linguistics and for giving us a chance to do a project of this nature. We also thank Mr. Siddharth Ranjan for his constant support both inside and outside the class in order to attend to our queries from time to time.

Introduction

In the late 18th century, the Independent Journal, the New York Packet, and The Daily Advertiser began publishing the so-called “Federalist Papers”, which were a compilation of essays written in order to promote the ratification of the United States Constitution. The authors of these essays were Alexander Hamilton, James Madison and John Jay, all of whom wrote under the pseudonym - “Publius”.

The federalist papers became the source of a very historical problem in the decades to follow because 12 of these essays had claims of authorship by both Hamilton as well as Madison. While some were able to correctly discern the identities of the authors of some of the essays, it wasn’t until Fredrick Mosteller and David Wallace[1] studied the problem with greater detail in 1962 that the identities of the authors were predicted with a great deal of certainty based on statistics. Their study demonstrated the power of Bayesian Inference in order to identify the authors of each essay.

We have tried to replicate some of their ideas by trying to identify authors of various novels. Our algorithm is based totally on the principle that every author has a distinctive frequency of words that is inherent to his or her writing style and this so-called unigram frequency is enough to predict the author of the novel. We have designed a simple algorithm to test this principle and we can identify up to 14 novelists.

In the later sections, we will also describe our study on how the era of a writer may influence the writing style and how that can affect the prediction of a novel whose author is not known. In addition to this, we have expanded these ideas to perform a bigram and trigram analysis as well. We have also tried to see if specific types of words (such as adverbs ending in “-ly”) can be useful to identify authors.

Objective

Our main objective is to identify the author of an unknown text. In this project, we’ve only focused on fiction books in the public domain, but this same algorithm, in theory, can be adapted to predict the authors of blogs, essays, news columns and so on.

[1] https://www.stat.cmu.edu/Exams/mosteller.pdf

Theory

We started by obtaining the top unigrams, bigrams and trigrams for the training data. Then, the unknown text's n-gram frequencies were compared to the n-gram frequencies of the authors' in the training data using absolute error. The probabilities were then calculated by subtracting the normalised error from unity. Then, an interpolated model[2] was created using weights 0.99,0.005 and 0.005 respectively, which basically told us how well the n-gram frequencies of the unknown text matched a given author from the training set. Lastly, the author with the least error and highest probability was then predicted to have written the unknown text. The weights were chosen so as to obtain the best results.

Dataset

We have a training dataset consisting of 98 novels written by 14 authors. The authors have been listed below:

Bram Stoker
Charles Dickens
Daniel Defoe
George Eliot
Haggard Rider
Henry James
Jane Austen
Jonathan Swift
Jules Verne
Lewis Carroll
Mark Twain
O Henry
Oscar Wilde
Rudyard Kipling

The novels have been downloaded from the Project Gutenberg website[3] in UTF-8 ".txt" format, and are freely distributable. The dataset we used can be downloaded from here[4].

[2] https://web.stanford.edu/~jurafsky/slp3/3.pdf
[3] https://www.gutenberg.org/
[4] https://github.com/AKnightWing/AuthID/archive/master.zip

The Algorithm

This method of identifying the author of a novel depends on the frequency of n-grams. For every author, we have calculated the frequency of all unigrams, bigrams and trigrams across all novels for that particular author. We have normalised this frequency by the total number of words across all novels for that particular author. This information is stored in a dictionary and is used for look-up during testing.
The same normalised frequencies are calculated for the test data sets as well for each novel separately.
Next, we calculate an error parameter as well as a maximum error for each author with respect to the test novel in the following manner:

For Unigrams:
Let x_1, x_2, x_3, ...x_n be the normalised frequencies of all unigrams for a particular author in the training dataset (across all novels written by him).

Let y_1, y_2, y_3, ...y_n be the normalised frequencies of all unigrams for the test novel whose author is unknown.

Here, each x_i and y_i refer to the frequency of the same word.

We hence calculate the error parameter as follows:

$$Error\ Parameter\ =\ \Sigma_i\ \left|y_i - x_i\right|$$

$$Max\ Error\ =\ \Sigma_i\ \left|y_i\right|$$

$$P_{unigram} = Probability\ Parameter = 1\ -\ \frac{Error\ Parameter}{Max\ Error}$$

We could select the author with the minimum error parameter as the most probable author, but alternatively, we choose to calculate a probability parameter $P_{unigram}$ as above.

Similarly, the error parameter, the max error and probability parameters P_{bigram} and $P_{trigram}$ are calculated for bigrams and trigrams as well. Finally, an interpolated probability parameter is calculated as follows:

$$P_{interpolated} = 0.99\ P_{unigram} + 0.005\ P_{bigram} + 0.005\ P_{trigram}$$

The author with the highest value of $P_{interpolated}$ is then selected as the most probable author.

Results and Accuracy

The interpolated model gave us the following results:
Of the 24 unknown texts, it predicted 21 of them accurately, which corresponds to an accuracy of 87.5 %.
The unknown texts were chosen randomly, and the results have been summarised in the following table:

Unknown Texts	Predicted Author	True Author	Result
A Tale Of A Tub	Daniel Defoe	Jonathan Swift	Wrong
All Around The Moon	Jules Verne	Jules Verne	Correct
Cleopatra	Haggard Rider	Haggard Rider	Correct
Eve's Diary	Mark Twain	Mark Twain	Correct
Hard Times	Charles Dickens	Charles Dickens	Correct
In The Year 2889	Jules Verne	Jules Verne	Correct
Intentions	Oscar Wilde	Oscar Wilde	Correct
Middlemarch	Middlemarch	Middlemarch	Correct
Oliver Twist	Charles Dickens	Charles Dickens	Correct
Rolling Stones	O Henry	O Henry	Correct
Roughing It	Mark Twain	Mark Twain	Correct
Sense And Sensibility	Jane Austen	Jane Austen	Correct
The Canterville Ghost	Jules Verne	Oscar Wilde	Wrong
The Fortunes and Misfortunes of the Famous Moll Flanders	Daniel Defoe	Daniel Defoe	Correct
The Gentle Grafter	O Henry	O Henry	Correct
The Innocents Abroad	Mark Twain	Mark Twain	Correct

The Life and Adventures of Robinson Crusoe	Daniel Defoe	Daniel Defoe	Correct
The Lifted Veil	Bram Stoker	George Eliot	Wrong
The Light That Failed	Rudyard Kipling	Rudyard Kipling	Correct
The Man	Bram Stoker	Bram Stoker	Correct
The Nursery, Alice	Lewis Carroll	Lewis Carroll	Correct
The People Of The Mist	Haggard Rider	Haggard Rider	Correct
The Turn of the Screw	Henry James	Henry James	Correct
Whirligigs	O Henry	O Henry	Correct

The 3 cases where the program was wrong are:

- A Tale Of A Tub, was written by Jonathan Swift. Our model predicted Daniel Defoe as the author. The fact that the model couldn't identify Jonathan Swift is not entirely surprising, because, Jonathan Swift's Training Data consisted of only 2 books, which is not enough. Thus increasing the data size should improve the result. A point to be noted is that both Jonathan Swift and Daniel Defoe were born in the 1660s, within a decade of each other. This fact prompted us to perform another analysis based on the era of the writer (details given in the next section).
- The Canterville Ghost, which has been written by Oscar Wilde. Our model predicted Jules Verne as the author. Again, both of these authors were from similar eras and thus the error could have been because of that. In the unigram and interpolated model, this was a wrong result, but the bigram model and trigram model gave us the correct result, showing that the algorithm was not wrong by a huge margin.
- The Lifted Veil, which has been written by George Eliot. Our model predicted Bram Stoker as the author. This result was intriguing and caught our attention, because neither of the unigram/bigram/trigram (and thus neither the interpolated model) were even close to correctly identifying George Eliot. The unigram and interpolated model tagged it as Bram Stoker, while the bigram and trigram model tagged it as Lewis Carroll. Thus this led us to believe that The Lifted Veil differs greatly from other works by George Eliot. And as expected, on further analysis, we found that this was indeed true. In fact, according to contemporary and modern critics, The Lifted Veil was least like George Eliot's style. George Eliot usually wrote realistic prose and poetry, not dealing much with the occult beliefs, such as clairvoyance. The Lifted Veil is the only one of Eliot's works that does deal heavily with these supernatural subjects.[5]

[5] Norton, Ingrid. "A Year with Short Novels: On Lifting Veils." Open Letters Monthly - an Arts and Literature Review. WordPress. Web. 07 Nov. 2010. https://www.openlettersmonthlyarchive.com/main-articles-page/short-novels-on-lifting-veil

Era of an Author

We notice that the “era” or the time period when an author has lived in has no effect on the identification process given a large enough dataset. The program that we wrote still distinguishes the authors correctly in most cases.

Henry James (1843-1916), Oscar Wilde (1854-1900), Bram Stoker (1847-1912), Haggard Rider (1856-1925) are all from approximately the same time period. We took Henry James’ novel as a test novel and completely removed Haggard Rider’s novels from the training set. We then ran our code to see if the program recognized any of the other 3 authors as a possible author for this Henry James’ novel.

The hypothesis here was that the era of an author might have some effects on the writing styles of an author.

This hypothesis was proven wrong by the algorithm as none of the 5 books of Henry James in our datasets were predicted to be written by Oscar Wilde, Bram Stoker or Haggard Rider.

We tried to do the same process with Oscar Wilde, Haggard Rider and Bram Stoker being the test datasets one after another. Once again, our hypothesis was proven wrong as there were a negligible number of positive results.

Hence, we see that the fact that the 4 authors are part of the 17th century has no direct affect on the n-gram usage frequencies and the identity of the authors thus predicted.

Role of -ly Adverbs:

In his book *Nabokov's Favorite Word Is Mauve*[6], Ben Blatt talks about how each author uses Adverbs differently. We thus decided to try using adverbs, specifically adverbs with the suffix “ly” to try and identify an author by linking each author with their adverb usage and then comparing the unknown text with this. We took care to not include -ly words which are not adverbs, the list of which was obtained from here[7].
However, the results were lacklustre. On running just the adverb model without the ngram model, we got an accuracy of just 12.5 %, that is, just 3 out of the 24 works’ authors were identified correctly. Thus, the frequency of -ly adverbs used is not a good characteristic of an

[6] Blatt, Ben. (2017). “*Nabokov's Favorite Word Is Mauve*”, Chapter 1, Simon & Schuster. ISBN:978-1501105388
[7] https://www.wordexample.com/list/ending-ly-not-adverbs?exclude_proper_nouns=1

author, and while it does tell us about the style of an author, this is not unique to an author. Thus this model was discarded. The archived program for this can still be obtained here[8].

Implications of our work and future explorations

It is indeed very strange that something as simple as unigram frequency can be such an accurate marker of an author. It is easy to print out the most common unigrams and see that articles (“a”, “an”, “the”), conjunctions such as “but”, etc. occur frequently in the top 5 most commonly used words. In common english, these are the words that are most often used to construct sentences. Therefore, we can probably say that the sentence structure of each author is distinctive and this distinction in sentence structure can be deciphered purely on the basis of the unigram usage of common grammatical words such as articles, pronouns and conjunctions.

And while the most frequent unigrams are common to different authors, their relative frequency is different and thus this can be considered as a characteristic of that author. The same can be said about bigrams and trigrams.

A point to be noted is that the unigrams, bigrams etc also included punctuations like fullstops, commas and quotation marks, whose frequencies are also unique to authors (for example, full stop frequency tells us about the average sentence length).

This algorithm can also be used to identify similarities between various writers based on sentence structure. One can employ the same method that was employed in the section on the era of an author. For example, if one wishes to predict which author one is closest to in terms of writing styles, all that needs to be done is to delete all the novels of that author from the training set and run the program with the novels of that author in the test set. Ideally, this should give us the author whose sentence structure matches closest to the one in question. In our experiment, we tried to check if the era of an authors had any effect on the identity of an author (or the sentence structure). We can check if there are other such aspects which can be related to sentence structure such as country, gender, genre, etc.

Similarly, it can also be used to identify the works of a single author which are unique and differ greatly from their other works, as in the case of The Lifted Veil. This can be done by modifying the algorithm such that only works by a single author are analysed, and then the one with most error or the least $P_{interpolated}$ could be classified as a text with a unique writing style.

Other applications where this could be used is to check claims of Pseudepigrapha, which are falsely attributed works, i.e, texts whose claimed author is not the true author.

[8] https://github.com/AKnightWing/AdverbAuthID/blob/master/adverb_predictions.py

A software with a similar but modified algorithm can also be released as a consumer tool, which could help users improve their writing skills and help write like professional authors.

The results can further be improved by increasing the size of the training set. Further, parameters like the POS tag frequency, sentence length, book length etc may be used as parameters for further models if required. And finally, all of these models can be mixed via interpolation, with suitable weights.

It is very impressive that something as simple as n-gram frequencies gave us an accuracy of 87.5%, and it would be really interesting to try and perfect this.