Authorship Attribution
Assignment 4 - Text and Multimedia Mining

Shaurya Gaur (s1116008)
Arina Schippers (s1036744)
Foteini Papadopoulou (s1122141)

November 15, 2023

1 Introduction

In this report, we will describe the design, results, and problems of working on the classifier for the authorship attribution problem. Using features, this classifier will try to determine who the author of a certain text is.

2 Features

In this section, we will provide an overview of the features that we selected to implement and use for the authorship attribution classification problem, what they express, the reason why we used them and what was the effect of each feature in the performance of our model.

Word count with TF-IDF transformation: The importance a word can be measured by Term Frequency - Inverse Document Frequency, or TF-IDF, transformation. The importance can be measured not only in a specific text, such as for the count vectorizer, but in the whole collection of documents. The TF-IDF transformation can capture word counts that indicate the text's topic and, next to that, understand the preference of an author to use certain words. We chose this feature since the word use of an author could tell a lot about them.

Character frequencies: The next feature we look at, is character frequencies by counting the number of occurrences of each character. We chose this because it has been shown to quantify the writing style of the author well.[1] We have decided not to lowercase the characters because we wanted to capture the preference of each author of a formal tone using capitalization or an informal tone using more frequent lower letters.

Average word length: Another lexical feature we are using is the average word length. This expresses the author's preference to use simple and short or complex and long words to convey their ideas.

Ellipsis count: Authors may convey some dramatic pause with '...', or just use them as a normal part of speech. After having a glance at the data set, we noticed multiple

[1] Inspired by lecture notes from the TXMM course, lecture on Authorship attribution

usages of ellipsis, so we decided to use it.
POS tag Counting: One group of syntactical features that we have selected is the Part-of-Speech (POS) Tagging Counts. POS tagging consists of assigning the part of speech to words in a text. It is used to analyze the structure of a text and unveil patterns that an author uses, for example, a high number of verbs or adjectives.
Stop words count: The stop word counting feature expresses the writing style of the author and according to the Sockpuppet paper [2], it is generally accepted that the frequency of stop words is a useful characteristic to identify authorship.
Gender Pronouns frequency: With this feature, we focus on counting the gender pronouns "he" and "she" in each author's text. As the Socketpuppet paper notes [2], authors may have a preference for using more of these pronouns. After glancing at the dataset, we noticed the grate usage of these pronouns.
Total number of sentences: Counting the number of sentences can depict the preference of an author to use multiple sentences and how they organize the text, according to the Socketpuppet paper mentioned above. We came up with this when looking at the texts in the dataset. There we noticed that the sentence length differs per author.

3 Classifier Selection

We evaluate the effectiveness of three classification models. First, we tried Random Forests (RF), using 100 estimators and a max tree depth of 7. We also examined a Multinomial Naive Bayes (NB) model with $\alpha = 1.0$, and a Support Vector Machine (SVM) with a regularization parameter of 1.0 and a radial basis kernel function. Table 1 illustrates that the RF classifier performs best in this task, performing significantly better than NB and SVM, especially when using all features.

Table 1: Performance of the three different models on the development set.

Model	**All Features**			**Best 100 Features**		
	Prec.	Recall	F1	Prec.	Recall	F1
RF	0.93	0.92	0.91	0.76	0.70	0.69
NB	0.63	0.53	0.50	0.65	0.53	0.53
SVM	0.40	0.52	0.44	0.40	0.52	0.43

Additionally, we test the impact of feature selection in this task. Our eight feature categories yield a total of 44,359 features, which can be computationally expensive for classification. For this, we use the `SelectKBest` selector ($K = 100$) from `scikit-learn`[1], which computes the ANOVA F-score for each feature. From the right-hand side of Table 1, we find that this type of feature selection did not yield improvements in the NB and SVM models, and cost a significant amount of performance on the RF model. This suggests that RF models are robust to this type of feature selection, though an ablation analysis is required.

4 Ablation Analysis

To determine which features have the greatest impact on classifying fanfiction authors, we run an ablation analysis on our eight feature groups. In this process, we leave each group of features out individually and run the classifier. Figure 1 visualizes that the *character counts* features have the greatest impact, since removing them yields the lowest F1 scores. On the other hand, *POS tags* yielded the highest F1 scores. Below, we analyze the relative importance of each individual feature.

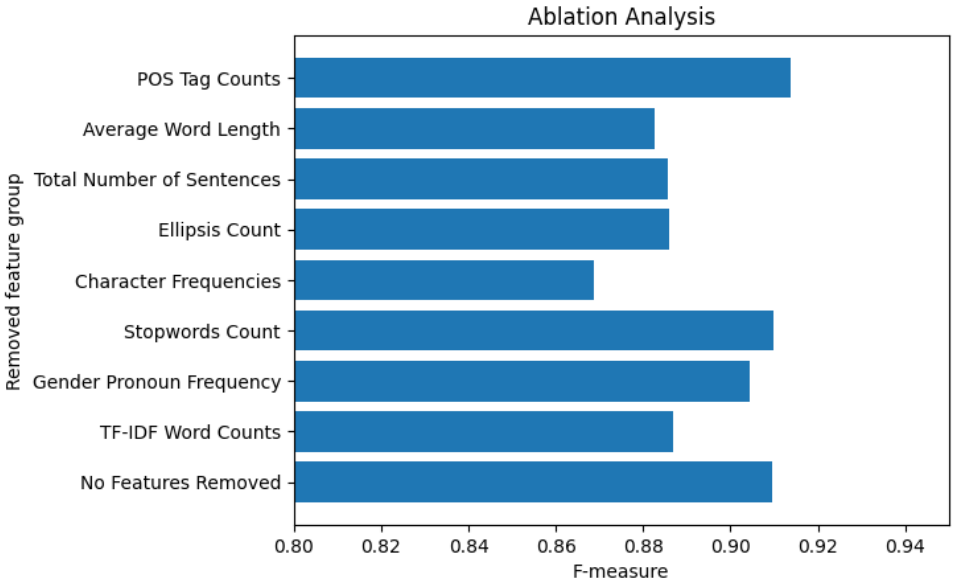

Figure 1: Histogram comparing F1 scores from our Random Forest classifier with specific feature groups removed.

Word count with TF-IDF transformation: According to figure 1, this syntactical feature is the least important and does not contribute as much as other features.
Character frequencies: Counting the character frequencies is demonstrated to be one of the most informative features as can be seen in the graph. The character frequencies express the writing style of an author and how long or short they made their texts. This seems to contribute significantly.
Average word length: This feature contribute the most after the character frequencies. This means that different authors have different word lengths. This explains more which author has written a text than every other feature except for character frequencies.
Ellipsis count: Based on the ablation analysis, we have seen that ellipsis contributes significantly to the overall performance of our authorship classification. This means that the way authors use punctuation differs a lot.
POS tag Counting: This feature was the least significant of all of the features. This means it is of less importance to distinguish the author.
Stop words count: The usage of stop words by authors did not seem one of the main important features, since it is among the features with the lowest impact on the f-score.
Gender Pronouns frequency: Based on the graph, we have noticed that gender pronouns frequency contributes slightly to improving the performance of our authorship classification problem.
Total number of sentences: Looking at the graph, it can be noticed that this feature

is relatively informative to classify an author to the given dataset. This feature performs on the same level as the ellipsis feature. This means that it does contribute to the classification problem, but not as much as others.

5 Results

Table 2 shows our Random Forest model's performance on the development and testing sets using all 44,359 features. From these results, we conclude that our model is able to attribute fanfiction authors well. The similarity between development and testing results indicates that this model generalizes well to this type of fanfiction.

Table 2: Performance of our Random Forest model on the development and test set.

Dataset	**Precision**	**Recall**	**F1**
Dev	0.93	0.92	0.91
Test	0.91	0.88	0.88

6 Discussion

We encountered a few key problems when developing our classifier. When extracting POS tags, we originally collect 44 features in the training data. However, we found a discrepancy in the development set, since it was unable to extract the `SYM` feature from our NLTK `pos_tag`[2] extractor when transforming that data. As a result, we omitted this feature from the data. Additionally, when training the RF model with default parameters, we noticed that it overfit to the training data, yielding F1 scores of 1.00. We discovered that if a maximum depth was not set, it would fit perfectly to training data, so therefore, we reduced it to 7.

Finetuning the parameters for the classifier and each of the feature extractors, adding more features, and selecting the appropriate amount of features would be done if we had 10 hours more to try out more parameters and test how our model performs on it. We would also add more features. The ones we are curious about are average sentence length and character names. Character names are relevant since authors might write a lot about the same characters. These might be different from other authors. We would also take a closer look at the dataset to see if we can find more patterns. We already looked at it and found, for example, the ellipsis, but there might be more.

In general, we found a few key difficulties with authorship attribution. Since dozens of types of features can be used [2], optimizing the collection of each can be intensive in time and resources. Moreover, identifying strong features may be domain specific, requiring either existing domain knowledge or time spent closely examining the dataset text. For example, in our case, we noticed the ellipsis usage in the given fanfiction texts, while the Sockpuppet authors noticed the frequencies of different grammatical errors.

[2]https://www.nltk.org/api/nltk.tag.pos$_tag.html$

References

[1] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[2] Thamar Solorio, Ragib Hasan, and Mainul Mizan. “A Case Study of Sockpuppet Detection in Wikipedia”. In: 2013. URL: https://api.semanticscholar.org/CorpusID:188976.