Stylometric Analysis of the Pentateuch using AI

Abstract

Modern biblical scholarship holds that the Pentateuch, also known as the Torah, is a multiauthor document that was composed over a period of hundreds of years. However, scholars disagree on the number and circumstance of the authors who have contributed to the Torah with some adhering to the older documentary hypothesis (DH) and many others prescribing to the newer, modern supplementary hypothesis (SH). This work aims to shed light on this controversy using Natural Language Processing (NLP) to identify the authors of the Torah at the sentence level. Computerized stylometric analysis in this piece reveals an intricate story showing the lack of a strong stylometric signature from the E source over the J source and a strong seepage of the P source into sources thought to be independent by the documentary hypothesis.

Introduction

The last decade has seen rapid advancements in broader AI, machine learning, and natural language processing. This is especially true when it comes to text embedding techniques, of which recent advances have made this analysis much more successful than if it would have been performed only a few years earlier. Authorship identification is a rapidly growing subset of NLP with a wide breadth of uses ranging from historical text scrutinization, plagiarism detection, all the way to user deanonymization on social networking sites.

It is with this context that we turn our guise to the long established field of biblical scholarship where the Hebrew Bible is analyzed under an academic and historical lens. In the 19th century the documentary hypothesis started to form. It was built upon gradually by various scholars and came to full fruition through the works of Karl Heinrich Graf and Julius Wellhausen. It argues that the Torah is a compilation of four originally independent documents written by four different authors over the span of 500 years before being complied by a final redactor (Getrz et al., 2016). According to the DH, J and E are the oldest sources (10^{th} or 9^{th} centuries BCE - disputed) and they influenced the creation of the D and P sources (Getrz et al., 2016). These four sources were then compiled and modified by a redactor, R, to make the Torah. This contrasts with the supplementary hypothesis which in its modern sense was developed through the works of scholars like John Van Seters, Rolf Rendtorff, and Hans Heinrich Schmid. The SH argues that the D source was written first around 600 BCE and was followed shortly thereafter by the J source and then P source (Van Seters, 1999). The most significant change being that there is no E source; they hypothesize that J may have included earlier works into their own (Van Seters, 1999). In this version of the SH there is no R source as well; it argues that P acts as J's redactor and thus there is no need for an R source (Van Seters, 1999). These are not the only hypotheses about the composition of the Torah, but they are the ones that have garnered the most attention.

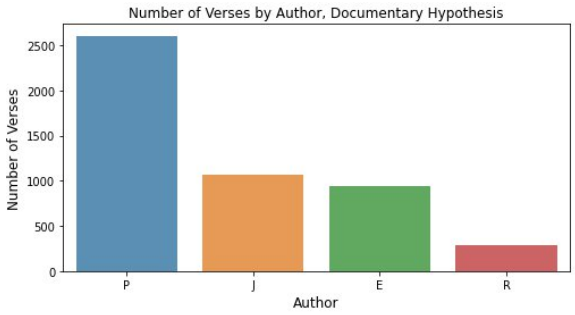

Figure 1.A. The number of verses attributed to each author according to the DH.

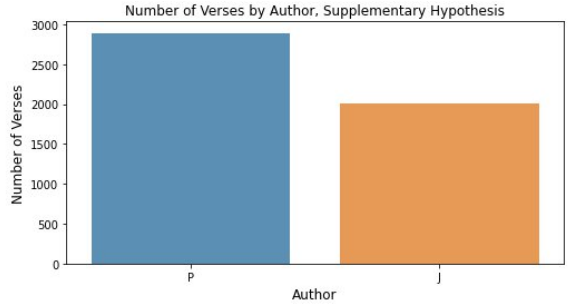

Figure 1.B. The number of verses attributed to each author according to the SH.

In this paper we hope to shed light on the documentary versus supplementary hypothesis controversy through the use of natural language processing (NLP). This authorship identification problem is a classic classification task, for which there is unsupervised classification and supervised classification. We will use both in our study but will rely heavily on supervised classification for our findings and analysis. It should be noted that many authorship attribution problems are not so lucky as to have a precurated set of labels that have been meticulously crafted by scholars over hundreds of years. Unsupervised authorship identification with our current embedding techniques and clustering algorithms tends to be less performant than supervised authorship identification, especially when picking the number of authors, k, is required.

Document size is a second factor critical to clustering success. In general, the larger each document is, the greater clustering performance becomes (Yu 2019). Clustering a set of paragraphs or even chapters will be more successful than clustering a set of sentences. In this study, we classify at the sentence level as this is the most appropriate resolution to view the Torah at because of its complex compositional history. This will impact performance but will provide us with more trustworthy results in the long run. There is a rather insignificant number of sentences – less than one hundred – that scholars have split in half, arguing that one half may be the work of one author and the second half the work of another. In such an event we have labeled the mixed sentence with the author scholars propose comes later as they are the final editor and have likely expressed their stylometric signature upon it.

The final factor critical to authorial identification is the text embedding technique used. This is significantly more important than the classification algorithm used. There are more or less two schools of thought when it comes to what data to provide to the text embedding algorithm. One such is to directly input lexical information which tends to be more performant. This would mean directly taking verses from the Torah and processing it as data. The alternative is to use a stylometric feature such as parts of speech (POS), which tends to be slightly less performant (Yu, 2019). This means that each document is converted from text to POS tags. For example, the sentence, “the dog runs in the park,” would become “Determiner Noun Verb Preposition Determiner Noun.” This method removes situational context from a given document while retaining authorial style. If one is trying to discover the underlying authors of a conjoined document, and one author is

consistently talking about taking their dog on a walk in a park, while the other is explaining a recipe on how to make the perfect jelly doughnuts, the classifier will likely key in on words like “dog” and “doughnut” and classify by context. This is not ideal for authorial identification tasks. Any sentence where contexts overlap, “I ate a jelly doughnut and my dog begged for a bite,” is bound to be misclassified by a lexical classifier. The situation would be made even worse in our case where our authors are writing about the same topics. In this study we will use POS embeddings instead of lexical embeddings and it is suggested that the reader look at any author identification task that uses lexical embeddings instead of stylometric embeddings with extreme caution and criticism.

POS embeddings may be even more beneficial for this particular context as the Tanakh as a whole contains some 1480 hapax legomena, or single use words, which would not be particularly beneficial in an author identification task. It should be noted that there are other stylometric feature sets that may work better than POS, but POS is what was available in Hebrew for this project (see Brian Yu’s *Stylometric Features for Multiple Authorship Attribution* for more details).

Method

We use hand curated POS tags from the OpenHebrewBible in conjunction with sentence level labels from tanach.us as our data source in this study. Tanach.us only provides sentence level labels for the DH, so the SH labels were created by converting all E labels to J and all R labels to P. As both the SH and DH agree that Deuteronomy is an almost entirely independent source (albeit dating it to different time periods), it will be omitted from classification. This is appropriate as each additional k only serves to increase the possibility for misclassification and because Deuteronomy from a scholarly standpoint is really more closely aligned with the other Deuteronomistic histories such as Joshua, Judges, Samuel, and Kings than it is to the other four books of the Torah (Getrz et al., 2016). The corpus used is exclusively POS tags collected from the first four books of the Torah. Pretrained Hebrew embeddings are not used in this study because they are trained on lexical data and because modern Hebrew has its differences from Biblical Hebrew. Various embedding techniques and classification algorithms were tested in order to obtain the best results. Our end goal is to determine if our classifiers can better fit our predicted DH or SH labels to their respective scholarly labels.

Unsupervised Classification

To start, we attempt an unsupervised classification method that will point us in the right direction, but will ultimately prove overly ambitious at the sentence level given the small size of our data set and the complexity of our task. We’ve chosen to include it as this method because many author identification tasks do not have a precurated set of labels and many do not even have a known number of authors, k. Future performance improvements could be obtained with either better stylometric feature sets, embedding techniques, and or classification methods.

We begin by taking our POS tags for all verses in the Torah except those in Deuteronomy and shuffle them randomly.

We then embed these sentences using various techniques like word2vec (W2V), fastText, term frequency inverse document frequency (TFIDF), and count vectorization. We apply an unsupervised classification algorithm to the data twice, once with k=2 for the SH and once with k=4 for the DH. For each classification run, we calculate k centroids for our data set where each data point is a verse converted to POS and then embedded. Any point within two standard deviations of its respective centroid is kept under the theory that it is a "core data point" that best represents that author's style (Daks & Clark, 2016). All data points outside those two standard deviations are reclassified using a supervised classification algorithm in which those points within the two standard deviations are used as labeled, true data. This technique and some code is taken from Alon Daks and Aidan Clark's (2016) paper, *Unsupervised Authorial Clustering Based on Syntatic Structure*. We will use spectral clustering with supervised improvement from a random forest classifier just as they did.

Supervised Classification

We begin by taking our POS tags for all verses in the Torah except those in Deuteronomy and shuffle them randomly. They are then split into 80% training data and 20% testing data. There are two sets of labels, one for the DH and one for the SH accordingly. These POS tags are then converted into various text embeddings such as w2v, fastText, TFIDF, and count vectorization. Glove and doc2vec embeddings were tried but they were so unsuccessful they were not worth including. These embeddings are then run through 5-fold cross validation with various classification algorithms such as multinomial naive bayes (MNB), random forest (RF), and logistic regression (LR), and support vector machines (SVM). As in the unsupervised classification study, the exact classification algorithm used is significantly less important than the embedding technique. SVM and RF will be used most in this study because they offer strong performance in a reasonable time.

Multiple embedding methods will be tested to obtain the best performance possible. Each embedding method will be run with 5-fold cross validation and will be run twice, once where k=2 for the SH and once where k=4 for the DH. The average log loss will be recorded to determine the success of the model. Minimizing log loss, our evaluation metric, will maximize overall learning. Truncated SVD data will be calculated as well in the hopes of capturing additional style (Rajkumar, 2018). The results from all the embedding techniques and SVD data will be compiled and run through 5-fold XGBoost classifier to achieve the best results possible. XGBoost will be run twice, once for boosting the SH models and once for the DH models. We will then plot a feature importance histogram to determine which models provided the most stylistic data to XGBoost as well as a normalized confusion matrix to verify how accurate our predicted clustering is compared to our true labels. As there are slight variations in data due to the random seed provided to the classifiers, the results presented will be the average of 25 runs of the entire process as described above. The basis of this technique and some of the code is modeled off of the *Simple Feature Engg Notebook - Spooky Author* on Kaggle. These are the final results that will be displayed in the study. It should be noted that fastText, a relatively new text embedding technique, is so effective that it classifies almost just as

well as the XGB classifier in a much smaller compute time.

Results

Unsupervised Classification

Our most successful embedding techniques used were the count vectorizer and TFIDF both with the character level analyzer. These embeddings were run through our unsupervised classification algorithm - spectral clustering - before receiving supervised improvement from a random forest classifier. Only data points that are more than two standard deviations away from a centroid were reclustered during the supervised improvement phase. The DH model with CV_char embeddings returned a weighted F1 score of 0.44 and the SH model received a weighted F1 score of 0.68. While this is not necessarily impressive, we see that the classifier is in fact picking something up some authorial style for both narratives. These results would be rather uncompelling if used to validate any one hypothesis over another, so we will move on to supervised classification.

Embedding	DH F1_Score	SH F1_Score
TFIDF	0.40	0.59
fastText	0.34	0.65
W2V_sg	0.32	0.58
W2V_cbow	0.38	0.61
CV	0.41	0.66
CV_char	0.44	0.68
TFIDF_char	0.43	0.68

Figure 3. The F1 scores from unsupervised classification (spectral clustering) with supervised improvement (random forest) with various embedding methods for both the DH and SH.

Supervised Classification

We start by testing out multiple embedding methods with supervised classification. Each embedding method is run through a 5-fold cross validation supervised classifier twice, once of the DH where k=4 and once for the SH where k=2. The predicted labels from these classification experiments are then recorded. As the classification algorithm used tends to have the least impact on performance but the most impact on run-time, we tried to use algorithms that find a middle ground between maximizing performance while still maintaining a reasonable run-time. The results of our many embedding techniques and classification tests are then run through XGBoost with 5-fold cross validation twice, once for the DH where k=4 and once for the SH where k=2. Our goal here is to minimize log loss. In both cases our most important feature came from fastText. This entire process is run 25 times and an average is calculated to mitigate any inconsistencies a particular

random seed may cause in the confusion matrices and overall results. We computed an average classification log loss of 0.70 for the DH and 0.33 for the SH.

We then plotted two normalized confusion matrices in which true labels would be compared against labels predicted by their respective classifiers. For the DH, we found that we could very easily identify the P source's unique stylomentric signature. We correctly classified 90% of the P source. This finding is paralleled by our SH model which correctly classifies 88% of the P source. The classification of our other sources, however, begins to create a more complex scene. In the DH model we only correctly classify 57% of the J source, with roughly 20% misclassification going to E and P respectively. This contrasts to the SH model in which we correctly classify the J source with 83% accuracy. At this point, misclassification becomes much more interesting than classification done correctly by the model. We see that the DH model correctly classifies E only 40% of the time and misattributes 36% of E to J and 25% to P. We also see that the DH classifier has trouble distinguishing between P and R with 53% attributed to R and 37% attributed to P. We then plot a new hybrid confusion matrix using the SH model but the DH labels. This tells us that 81% of sources traditionally attributed to E are classified as J in the SH model and 88% of what is traditionally attributed to as R is now classified as P by the SH model.

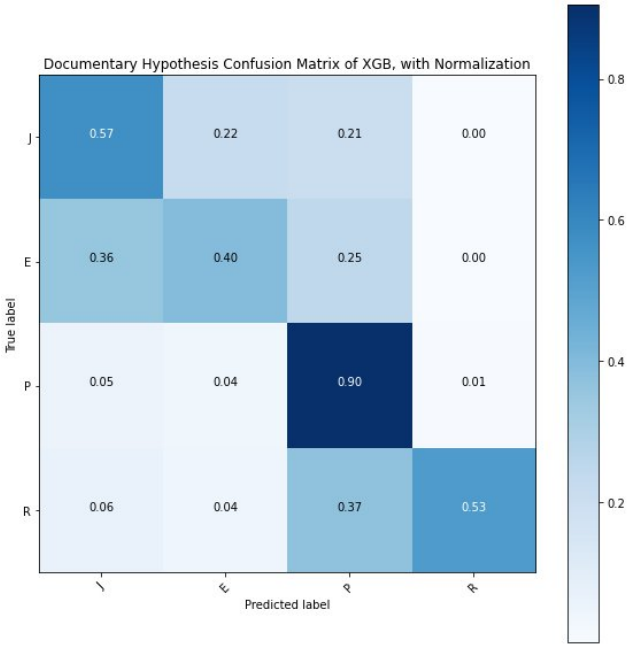

Figure 4.A. Normalized confusion matrix of the DH XGBoost model. Average of 25 runs.

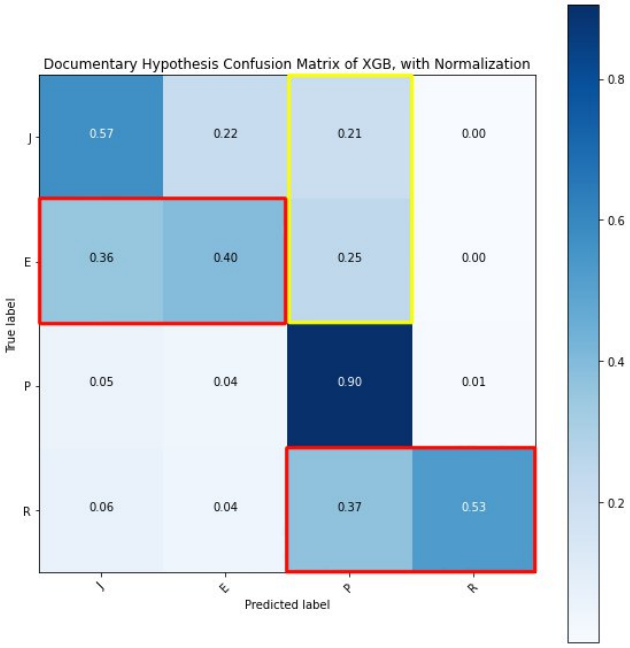

Figure 4.B. Normalized confusion matrix of the DH XGBoost model. Red boxes indicate two proposed authors that the classifier struggles to distinguish between (J/E and P/R). Yellow box shows the seepage of P's style into the works traditionally attributed to J/E. Average of 25 runs.

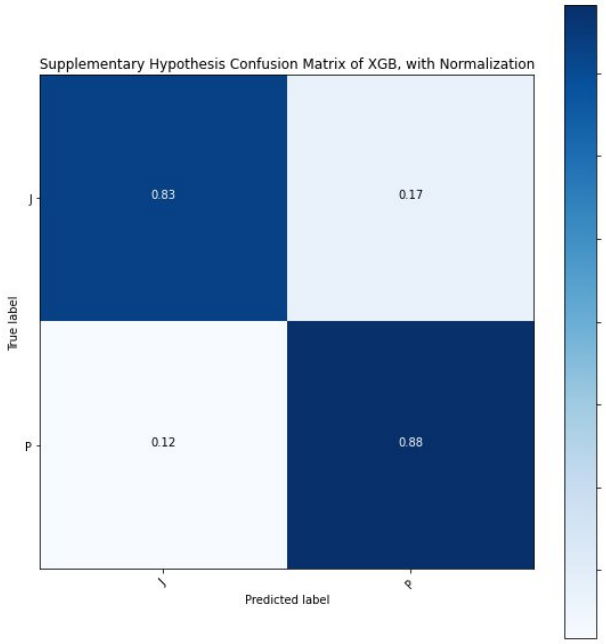

Figure 4.C. Normalized confusion matrix of the DH XGBoost model. Average of 25 runs.

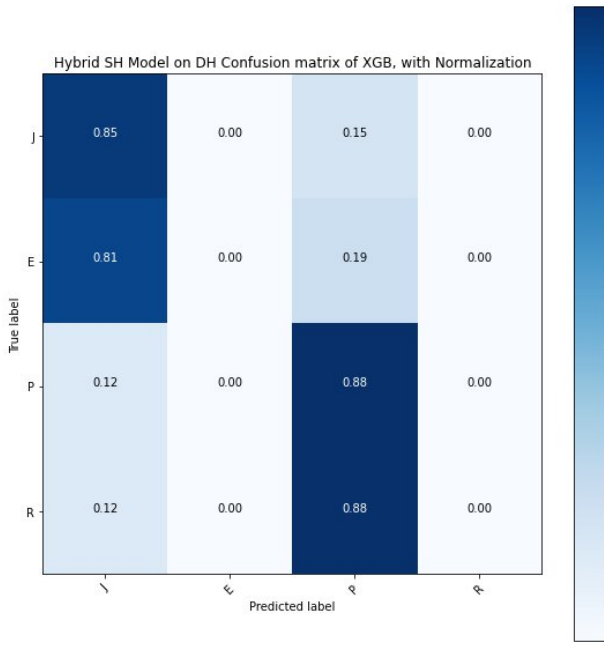

Figure 4.D. Normalized hybrid confusion matrix of the SH XGBoost model plotted on the DH confusion matrix. Most verses attributed to E by DH scholars get classified as J in the SH model. The same is seen with R being classified mainly as P. Average of 25 runs.

Conclusion

The Pentateuch is an ancient document with a rich and complex history. In this analysis we sought to find if the documentary hypothesis or the supplementary hypothesis better fit our model based on stylometric analysis. We found that we were able to fit the SH with a lower log loss and greater degree of accuracy than we were the DH. In addition, we found that our DH classifier struggled to distinguish between J and E. When our SH classifier was plotted under a DH confusion matrix, we found that the vast majority of verses the DH model attributed to E are reassigned to J in the SH model. This shows that there is no strong stylometric basis for the belief in an E source. If there ever was an E source(s), its style has been edited by the J source so much that essentially no unique stylometric signature remains. In addition, the DH classifier frequently mislabels J and E verses as P, at a rate of roughly 20%. This is perhaps the most interesting finding of the entire work as it shows that P has most likely implanted their stylometric signature on a significant portion of writing that has been traditionally thought to be J/E's. This is evident in the SH classifier as well, where there is a slight increase in J misclassifications for P. The DH classifier struggles to classify R with roughly half of predicted classifications going to P and to R. When we look at the SH classifier in the DH confusion matrix we see that 90% of these labels are classified as P. This shows that there is no strong stylometric basis for the belief in a single large redacting source.

These findings are therefore not in agreement with the documentary hypothesis. There is no clear evidence for a substantial E or single large R source as claimed and there is evidence of significant

perforation of the P source into sources previously thought to be J/E. As a takeaway, the authorship of the Torah can best be thought of as "J++,P." We see that the Torah is written by J, who (from a lexical understanding only) likely drew from what is now a shadow source(s) and left no significant trace of their stylometric signature. This work of J+ was then heavily modified by the P source giving us J++. Then P adds in their own complete contributions to what became the Torah, giving us J++,P. While this is a strong start, the documentary versus supplementary hypothesis debate hardly scratches the surface in the field of biblical scholarship. With an overall classification accuracy of only 86% on the SH model, there very well could be more to the story that is yet to be uncovered as new hypotheses are proposed and more advancements in NLP and broader AI are made in the near future.

References

Daks, Alon and Clark, Aidan (2016). *Unsupervized authorial clustering based on syntactic structure.* Computer Science Division, University of California, Berkeley. https://aclanthology.org/P16-3017.pdf

Gertz, J. C., Levinson, B. M., Rom-Shiloni, D., and Schmid, K. (Eds.) (2016) *The Formation of the Pentateuch: Bridging the academic cultures of Europe, Israel, and North America*. Mohr Siebeck Tubingen.

Rajkumar, Sudalai (2018). *Simple feature engg notebook – spooky author.* Kaggle. https://www.kaggle.com/sudalairajkumar/simple-feature-engg-notebook-spooky-author

Tanach. Tanach.us. Retrieved May 4, 2021, from http://www.tanach.us/Tanach.xml

Van Seters, John (1999). *The Pentateuch: A social-science commentary.* Sheffield Academic Press.

Wong, Eliran (2019, October 1). *OpenHebrewBible.* GitHub. https://github.com/eliranwong/OpenHebrewBible

Yu, Brian (2019). *Stylometric features for multiple authorship attribution* [Bachelor's thesis, Harvard College]. Digital Access to Scholarship at Harvard. https://dash.harvard.edu/bitstream/handle/1/37364618/YU-SENIORTHESIS-2019.pdf?sequence=1