Abby Kim
May 16, 2022

How Nasty Was Nero Really? A Computational Investigation

Introduction

Nero is regarded as the most infamous Roman emperor. Historical accounts cement his notoriety for slaughtering his loved ones, burning the city of Rome, and extreme tyranny. However, Rachel Mead reports in her *New Yorker* article "How Nasty Was Nero Really?" that a 2021 show at the British Museum offered a less sensationalist account of Nero's reign since modern scholars recognize that many accounts of Nero's atrocities closely resemble literary accounts of mythical events. In other words, the evils often associated with his reign might be grossly exaggerated for the sake of story and propaganda. (2021)

I want to investigate the notion that the primary Latin sources on Nero more closely resemble myth than history from a computational standpoint. To do so, I created a classification neural-network model that determines if Latin text is historical or mythological using the Latin BERT language model. I then feed in primary Latin text sources on Nero, so the model can determine if the literature is more mythological or historical.

Background Reading

The model uses Latin BERT, a contextual language model for the Latin language. The paper "Latin BERT: A Contextual Language Model for Classical Philology" explains how this language-specific model serves as a tool in both natural language processing for Latin and in using computational methods for traditional scholarship. (Bamman, D., & Burns, P. J., 2020) I use Latin BERT as a feature extractor that takes an input sentence and outputs a vector representation of that sentence. These features serve as the input for the classification model.

The introduction to the book *The Emperor Nero: A Guide to the Ancient Sources* provides context for Nero's birth and reign. It outlines the primary literary sources contemporary historians use to uncover the story of Nero. There are three main historians through whom we know Nero: Tacitus, Cassius, and Suetonius. First, Tacitus' *Annals* gives the most robust account of Nero's personal and political life. Cassius Dio's *Roman History* was written in Greek and will be omitted from this study. Suetonius' *Lives of the Caesars* is a biography that explores Nero's individual habits more so than his political impact. (Barret, et al. , 2016) The model will classify sentences in these texts as either historical or mythological.

Data and Methods

Preparing the Data

For the training data, Wellesley Professor Debra Freas provided insight into Latin historical literature and mythological literature. She mentioned Caesar's *De Bello Gallico*, Livy's *Ab Urbe Condita*, and Sallust's *The Histories*, as quintessential Latin historical texts. The primary accounts of Nero and the historical literature she mentioned are all written in prose. Much mythological literature, however, is written in elegy, a Latin poetic style. She recommended Ovid's *Metamorphoses*, Ovid's

Fasti, and Vergil's *Aeneid* as Latin mythological literature (all of which is elegy). To prevent the classifier from determining elegy vs prose instead of history vs myth, I also included mythological prose in the dataset. Prof. Freas recommended Petronius' *Satyricon*, Apuleius' *Cupid and Psyche*, and Apuleius' *Metamorphoses* (all of which is prose).

For the testing data, I used the two Latin primary texts on Nero: Tacitus' *Annals* books 12-16 and Suetonius' *Lives of the Caesars* chapters 61-63.

These texts are available online at *The Latin Library* (Carey). For some texts, I manually copied the Latin into a text file. For other texts that were split across multiple webpages, I created a script that scrapes each text and outputs the Latin into a text file. I then cleaned the text so only words (i.e. no numbers or non-wordy characters) remained in an all-lowercase, uniformly-spaced format. I removed all punctuation since punctuation does not exist in original Latin texts. (It is often added later by scholars for modern readability.) The cleaned text resides in 3 folders: "history," "myth," and "nero". Each "clean.txt" file consists of all the cleaned Latin text from the genre of its respective folder.

Example of raw Latin text file (excerpt from "data/history/caesar.txt"):

```
prohibent aut ipsi in eorum finibus bellum gerunt.

[Eorum una, pars, quam Gallos obtinere dictum est,
initium capit a flumine Rhodano, continetur Garumna
flumine, Oceano, finibus Belgarum, attingit etiam ab
Sequanis et Helvetiis flumen Rhenum, vergit ad
septentriones.

Belgae ab extremis Galliae finibus oriuntur,
pertinent ad inferiorem partem fluminis Rheni,
spectant in septentrionem et orientem solem.

Aquitania a Garumna flumine ad Pyrenaeos montes et
eam partem Oceani quae est ad Hispaniam pertinet;
```

Example of cleaned Latin text file (excerpt from "data/history/clean.txt"):

```
eorum finibus bellum gerunt eorum una pars quam gallos
obtinere dictum est initium capit flumine rhodano
continetur garumna flumine oceano finibus belgarum
attingit etiam ab sequanis et helvetiis flumen rhenum
vergit ad septentriones belgae ab extremis galliae
finibus oriuntur pertinent ad inferiorem partem
fluminis rheni spectant septentrionem et orientem
solem aquitania garumna flumine ad pyrenaeos montes et
eam partem oceani quae est ad hispaniam pertinet
spectat inter occasum solis et septentriones apud
helvetios longe nobilissimus fuit et ditissimus
orgetorix messala et p pisone consulibus regni
cupiditate inductus coniurationem nobilitatis fecit et
civitati persuasit ut de finibus suis cum omnibus
copiis exirent perfacile esse cum virtute omnibus
praestarent totius galliae imperio potiri id hoc
```

To prepare the data for the model, I chunked the text into 10 word segments that serve as a sentence unit. I then fed each sentence into Latin BERT to extract its sentence representation. A sentence representation consists of a 768-dimensional vector for a [CLS] (classification) token, each token in the sentence, and a [SEP] (separate) token. Since I am using Latin BERT as a feature extractor for my model's input, I excluded the vectors for the [CLS] and [SEP] tokens. I then created dataframes where each row consists of the Latin sentence, an indication of whether it is history or myth (this column is irrelevant for the Nero data), and a 7680-dimensional vector. The dataframe column "isHistory" is a 1 if the sentence is from a historical text and a 0 if the sentence is from a mythological text. The feature vector is of dimension 7680 since each word is represented by a 768-dimensional vector and there are 10 words per sentence. For some sentences, the tokenizer splits a word into multiple tokens, which would yield a sentence representation of higher dimension. If this is the case, I truncated the vector to include only the first 7680 features since the model requires all inputs to be of identical shape.

Example of described dataframe:

```
df.head()
```

	latin	isHistory	0	1	2	3	4	5	6	7	...	7670	7671	7672	7673
0	adepta loci grata domus cereri multas ea possi...	0	0.084854	-0.663267	-0.072145	0.090312	0.386681	-0.006475	0.031172	0.454453	...	-0.962970	-0.029183	0.210760	0.448808
1	solebam ipsumam meam debattuere ut etiam domin...	0	-0.396994	-0.230579	-0.090410	0.325092	0.725816	-0.497330	-0.004662	-0.449029	...	-0.657259	-0.109117	-0.192694	0.449039
2	bibit latenter abscondere et cum primum mitiga...	0	0.029655	0.121890	0.633323	0.504249	0.523247	-0.704886	0.600135	-1.150470	...	-0.612930	0.020685	0.764497	-0.101828
3	multa flavus harena mare prorumpit variae circ...	0	0.453690	-0.443973	0.159373	-0.224724	-0.014354	-0.300509	-0.387446	0.208028	...	0.125260	-0.428510	-0.408143	0.261395
4	quoque magna manent regnis penetralia nostris ...	0	1.079177	-0.745411	-0.718408	-0.823262	0.664935	0.406097	0.131723	-0.178671	...	0.021938	0.957372	0.650319	-0.175191

5 rows × 7682 columns

The historical data consists of 69,235 sentences, and the mythological data consists of 30,464 sentences. Due to memory constraints, I randomly selected 10,000 sentences from each dataset to serve as the training and validation sets. The Nero data consists of 2,338 sentences, so I did not need to simplify this sect of data for memory purposes. I saved the dataframes in different csv files consisting of 1000 rows each.

Building the Model

The model has the following architecture:

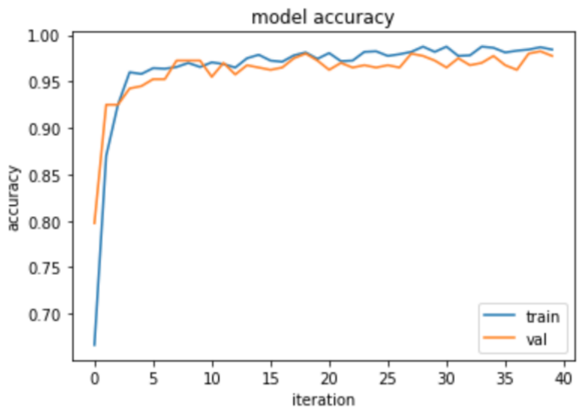
```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_3 (InputLayer)        [(None, 7680)]            0

 dense_8 (Dense)             (None, 4)                 30724

 dense_9 (Dense)             (None, 4)                 20

 dense_10 (Dense)            (None, 4)                 20

 dense_11 (Dense)            (None, 2)                 10

=================================================================
Total params: 30,774
Trainable params: 30,774
Non-trainable params: 0
_________________________________________________________________
```

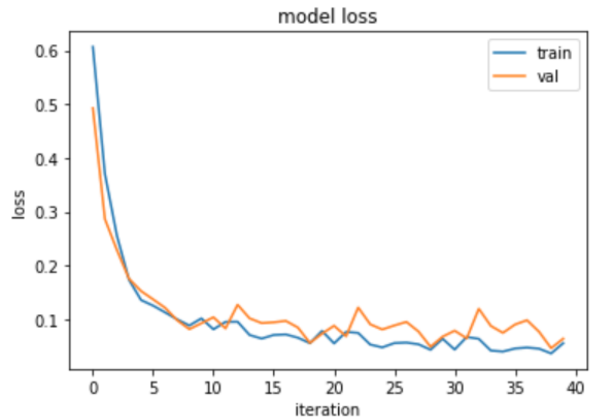
The input of the model is the 7680-dimensional sentence representation previously described. The model has 3 dense layers and a final softmax layer for classification.

Since the training data is separated into different dataframes, I used an unconventional method for training the model. The training data consists of 20 csv files. Each csv file represents a dataframe with 1000 rows. Half of these files consist of exclusively historical sentences, and half of these files consist of exclusively mythological sentences. Each iteration of training involved combining one historical dataframe and one mythological dataframe, shuffling the data together, then feeding it through the model. I used a batch size of 128 and a training-validation split of 80-20. An epoch involves feeding through all 20 csv files, 2 at a time, through the model. I trained the model for 4 epochs.

Model Data:

	loss	accuracy	val_loss	val_accuracy
0	0.606607	0.666458	0.492772	0.7975
1	0.370821	0.869837	0.286771	0.9250
2	0.255416	0.924906	0.228887	0.9250
3	0.173635	0.959950	0.175699	0.9425
4	0.136198	0.958073	0.152872	0.9450
5	0.125784	0.964330	0.137450	0.9525
6	0.113120	0.963705	0.121657	0.9525
7	0.100156	0.965582	0.098569	0.9725
8	0.088897	0.969962	0.082117	0.9725
9	0.102111	0.965582	0.093073	0.9725
10	0.081861	0.970588	0.104288	0.9550
11	0.095675	0.968711	0.083457	0.9700
12	0.095981	0.964956	0.127502	0.9575
13	0.071232	0.974969	0.102187	0.9675
14	0.064575	0.978723	0.093411	0.9650
15	0.071367	0.972466	0.094899	0.9625
16	0.072374	0.971214	0.097504	0.9650
17	0.066229	0.978098	0.085163	0.9750
18	0.056068	0.981227	0.056535	0.9800
19	0.078886	0.974343	0.073742	0.9725
20	0.055639	0.980601	0.088503	0.9625
21	0.077091	0.971840	0.068851	0.9700
22	0.075157	0.972466	0.122321	0.9650
23	0.053456	0.981852	0.091161	0.9675
24	0.048257	0.982478	0.081565	0.9650
25	0.056243	0.977472	0.089015	0.9675
26	0.057166	0.979349	0.095640	0.9650
27	0.054038	0.981852	0.077820	0.9800
28	0.043824	0.987484	0.049412	0.9775
29	0.064359	0.981852	0.068224	0.9725
30	0.044233	0.987484	0.079315	0.9650
31	0.067589	0.977472	0.065052	0.9750
32	0.064162	0.978098	0.120071	0.9675
33	0.042342	0.987484	0.088182	0.9700
34	0.040519	0.986233	0.075298	0.9775
35	0.046187	0.981227	0.090725	0.9675
36	0.047958	0.983104	0.098804	0.9625
37	0.045851	0.984355	0.077215	0.9800
38	0.037036	0.986859	0.046787	0.9825
39	0.056107	0.984355	0.064436	0.9775

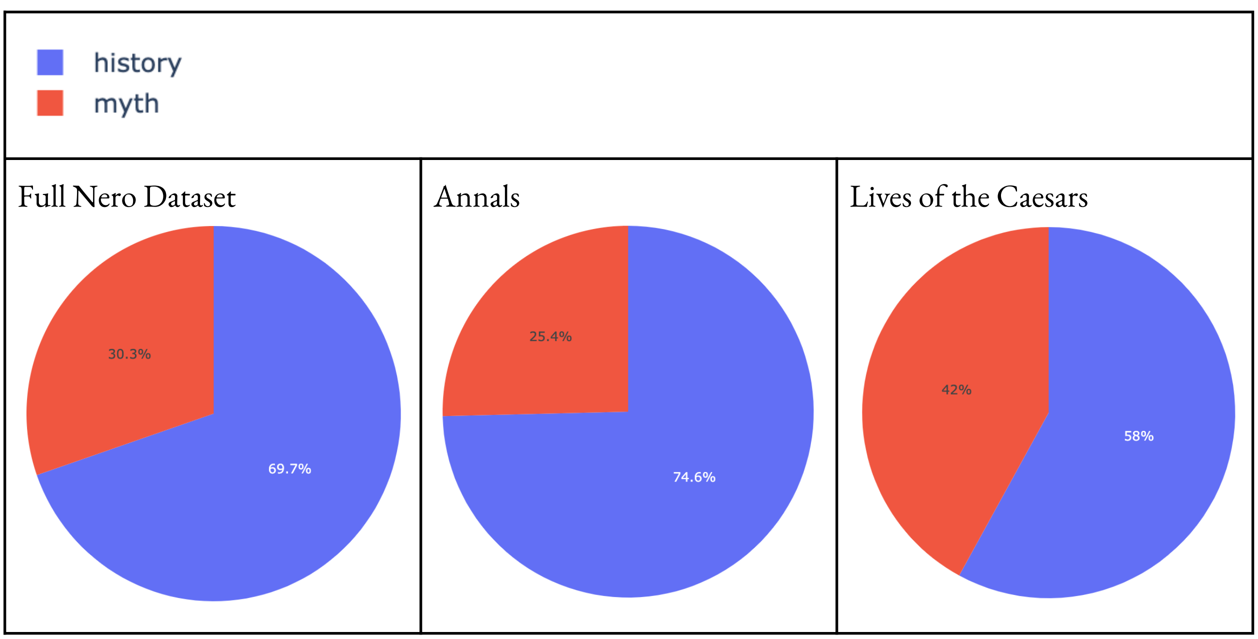




Results

The trained model then classified the 2,338 sentences from the Nero dataset. It classified 1628 sentences as historical and 708 as mythological. In other words, the model determined the Nero dataset to be 69.6% historical.

I then investigated each work in the Nero dataset individually. Of the 1573 sentences from Tacitus' *Annals,* the model classified 1174 as historical and 399 as mythological. In other words, the model determined the *Annals* dataset to be 74.6% historical. Of the 766 sentences from Suetonius' *Lives of the Caesars*, the model classified 444 as historical and 322 as mythological. In other words, the model determined the *Lives of the Caesars* dataset to be 58.0% historical.



Evidently, the model classified most of the Nero data as historical rather than mythological. About a quarter of the portion of the *Annals* dedicated to discussing Nero is classified as mythological.

Thus, instead of treating this work as historical non-fiction, classicists should study it under the context that Tacitus may have added linguistic flourishes popular in Roman mythological literature.

The model classified almost half of Suetonius' *Lives of the Caesars* as mythological. The higher proportion of mythological sentences compared to the *Annals* is unsurprising since Suetonius chose to focus on Nero's personal life rather than his political affairs. Latin historical texts tend to focus more on political affairs, while Latin mythological texts tend to focus more on personal relationships. Thus, when studying Nero through Suetonius, classists should be wary of taking the text as a strict historical recounting.

These results support the notion that some primary Latin text sources on Nero reflect mythological text more so than historical text. In other words, these computational conclusions reflect the conclusions drawn by traditional classical scholarship in this area of research.

Conclusion

In this paper, I introduced a neural-network that classifies a Latin sentence as either historical text or mythological text. I then used this model to investigate whether primary Latin sources on Nero more readily reflect historical or mythological language.

As in any empirical work, the robustness of the results depends on the data sources. The limited range of works included in the training corpus may have skewed the results in unknown ways. Since a neural-network is a blackbox, there is no way of knowing what the model is learning from the text. Thus, combining this computational approach with more traditional research techniques would allow for greater nuance when discussing the output of the model. For example, reading the specific sentences the model classifies as myth would allow a classicist to analyze what the model is recognizing and intuit which Roman myths the language of the sentence reflects.

While this paper utilizes Latin BERT to investigate a specific research question, I believe it outlines a computational methodology that compliments traditional classical research techniques. Traditional classical research relies on the intuition of the researcher to draw conclusions and find patterns across texts. A neural-network can detect patterns unforeseen by human researchers. Thus, it can serve as a guide for investigation. It can also serve as another form of validation for conclusions drawn in traditional research.

References

Bamman, D., & Burns, P. J. (2020, September 21). Latin Bert: A contextual language model for classical philology. arXiv.org. Retrieved May 14, 2022, from https://arxiv.org/abs/2009.10053

Barrett, A. A., Fantham, E., & Yardley, J. C. (2016). In The emperor nero: A guide to the ancient sources (pp. xi-xxiii). introduction, Princeton University Press.

Carey, W. L. (n.d.). The latin library. The Latin Library. Retrieved May 15, 2022, from https://www.thelatinlibrary.com/

Mead, R. (2021, June 14). How nasty was nero, really? The New Yorker. Retrieved May 14, 2022, from https://www.newyorker.com/magazine/2021/06/14/how-nasty-was-nero-really

Further possibilities:

- Close reading of actual sentences classified as myth vs historical
- Validate classification model with other novel texts
- More memory/resources - train the model on the whole corpus