A text analysis of Nietzsche's *Antichrist*

G. Palafox

September 14, 2020

Abstract

In the following report, we show the results of employing basic text analysis techniques on Friedrich Nietzsche's book *The Antichrist*. Graphics are shown to aid with the exposition of our findings.

1 Introduction

A text analysis of Nietzsche's *The Antichrist* [3] was done. We were able to find the most common words and characters in the text, omitting numbers and really common words (e.g., "the"). We used these words and characters to create various barplots and a word cloud visualization. Furthermore, we determined the most common pairs of words occurring together in the text, and show them with a network representation of our book.

2 Text Analysis

The text extraction and analysis was performed in a Jupyter[2] notebook running R[4] version 4.0.0[1]. We downloaded the book directly from Project Gutenberg's site using R's `gutenbergr` library. The book downloaded starts with an introduction by the translator, which we ommitted from the analysis, as the intention was to study the author's words. Our first step into analysing the text was to get the most frequent characters and words. For this we ommitted numbers, punctuaction, and so-called stop-words. Table 1 shows the ten most used letters and ten most used words in the text. Additionally, frequency of characters and words are shown in the barplots of Figure 1. For illustration purposes, we show a word cloud of the most frequent words in Figure 2. In a word cloud, the size of each word is proportional to the number of times it appears in the text.

Table 1: Word and character frequency

(a) Character frequency

	Letter	Frequency
1	e	15765
2	t	12898
3	i	10155
4	a	10052
5	o	9902
6	s	9114
7	n	9040
8	h	7654
9	r	7061
10	l	5291

(b) Word frequency

	Word	Frequency
1	god	172
2	life	96
3	christian	90
4	christianity	89
5	world	78
6	truth	54
7	priest	49
8	sort	48
9	instinct	47
10	power	46

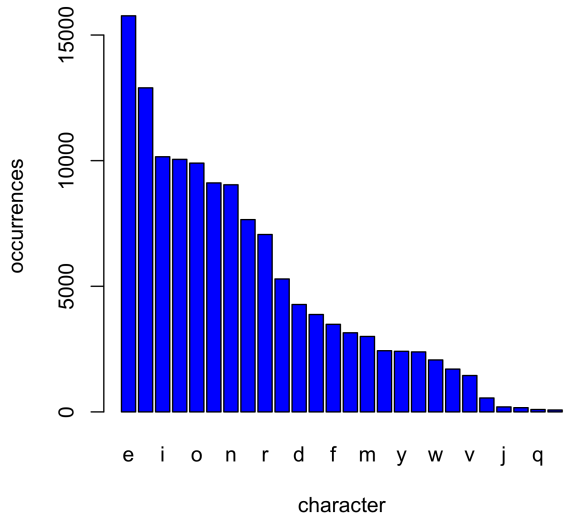
2.1 Network representation

Next, we created a network representation of our text. For readers unfamiliar with network (or graph) theory, basic definitions can be found in Appendix A. For our analysis, we considered words in the text as our vertices, joining a pair of words with an edge if they appeared together in the text (that is, if they form a bigram). Notice that since every edge joins exactly two words that appeared together in the text, there is a direct correspondence between our edges and the bigrams in the text. We made this a weighted network by assigning to each edge a weight equal to the number of times its corresponding bigram appears in the text. We restricted ourselves to those bigrams appearing more than once. The

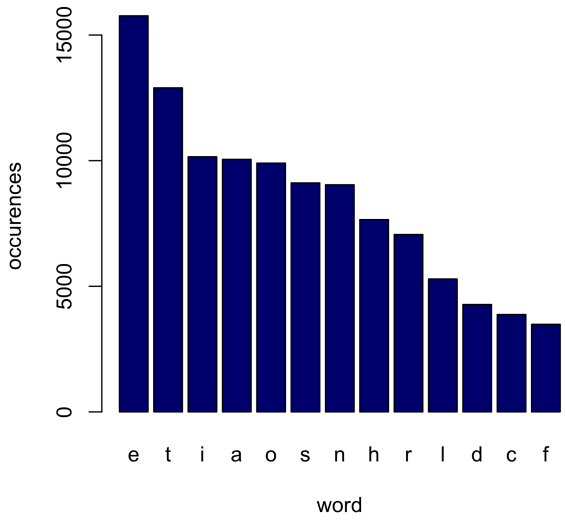
[1]The script and a Jupyter [2] notebook showing how we performed the data analysis and created the graphics in this report can be found at `https://github.com/palafox794/AppliedProbabilityModels/tree/master/Assignment2`

Figure 1: Barplots of words and characters occurences.

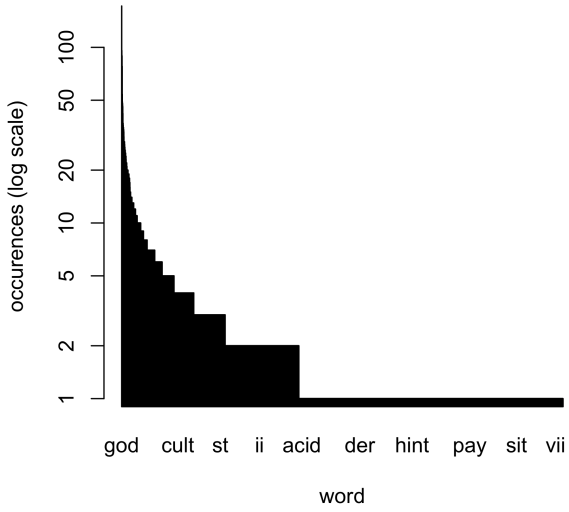
(a) Character occurrence.



(b) Most-frequent character occurrence.



(c) Word occurrence.



(d) Most-frequent word occurrence.

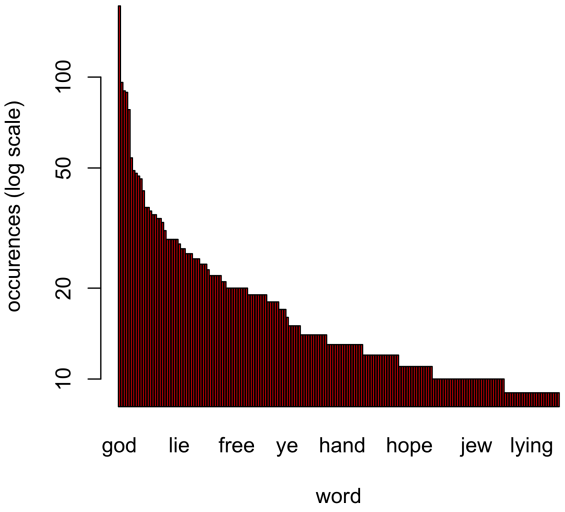


Figure 2: Word cloud of the book.

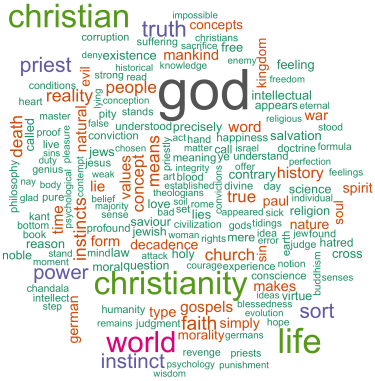
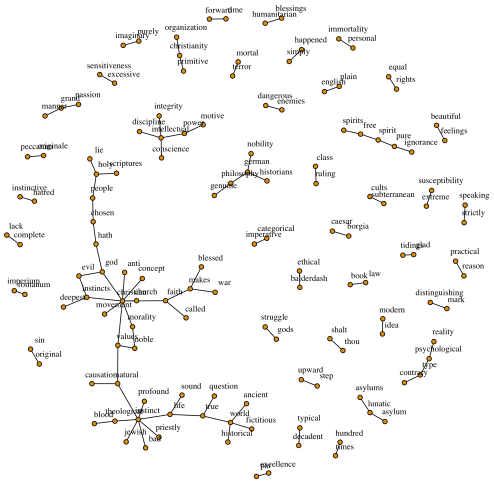


Figure 3: Book network representation.

(a) Network with words as vertices and edges joining words appearing as a bigram in the text.



(b) Same network, with vertex size and edge width proportional to their degree and weight.

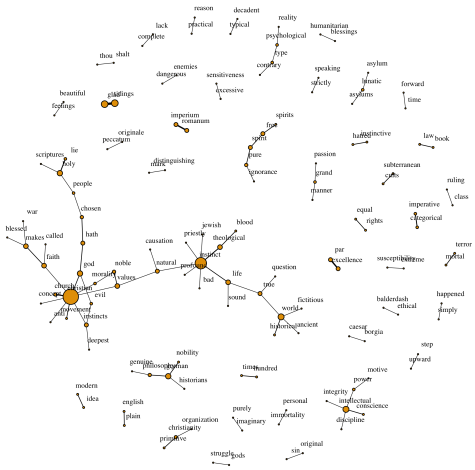


Table 2: Highest degree and strength vertices for largest connected component in the network.

(a) Sorted by degree.

vertex	degree	strength
christian	9.00	23.00
instinct	7.00	17.00
world	4.00	9.00
holy	3.00	8.00
faith	3.00	7.00
god	3.00	8.00
life	3.00	7.00
makes	3.00	7.00
natural	3.00	6.00
true	3.00	6.00

(b) Sorted by strength

vertex	degree	strength
christian	9.00	23.00
instinct	7.00	17.00
world	4.00	9.00
holy	3.00	8.00
god	3.00	8.00
theological	2.00	7.00
faith	3.00	7.00
life	3.00	7.00
makes	3.00	7.00
values	3.00	7.00

results can be seen in Figure 3. We also extracted the largest connected component of the network, as can be seen in Figure 4. Vertices with highest degree and strength can be seen in Table 2. Finally, we computed the degree distribution of both the whole network and of the largest connected component. The degree distribution gives us the relative frequency of n-th degree vertices, with $n = 0, 1, \ldots, \max_v\{\deg(v)\}$. You can observe these in Figure 5.

3 Conclusion

In accordance to what was expected given the theme of the book, the words *god*, *life*, *christian* and *christianity* appear the most in the text. The word *christian* is also the highest-degree word in our network. It is of interest to observe that while *instinct* is the 9th most appearing word, is the second word with highest degree (and strength) in our network. This means that, while overall is not the most used, it is still very "central"in joining words together. These are all very elementary findings. More sophisticated techniques can still be used to get a deeper study of Nietzsche's work, e.g., sentiment analysis. Further work can involve the comparison of different Nietzsche's books, or comparison of Nietzsche's books with works of other authors, in an attempt to characterize his writing style.

Figure 4: Largest component of network in Figure 3, with vertex sizes and edges width proportional to their degree and weight.

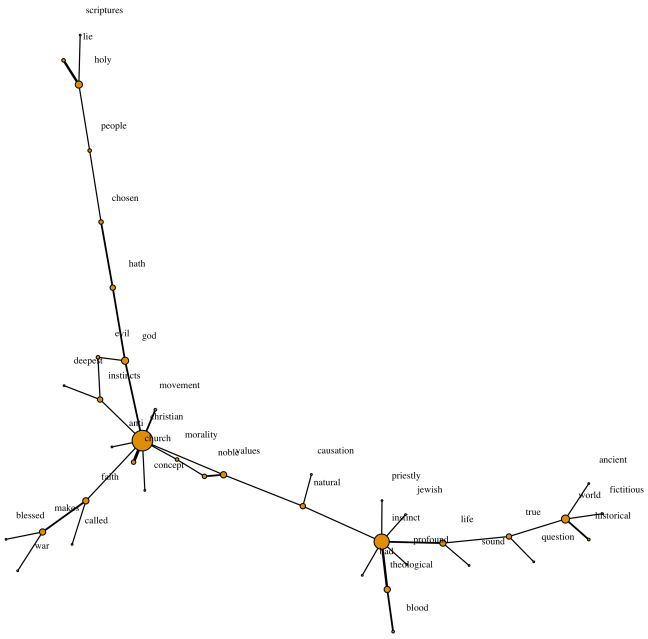
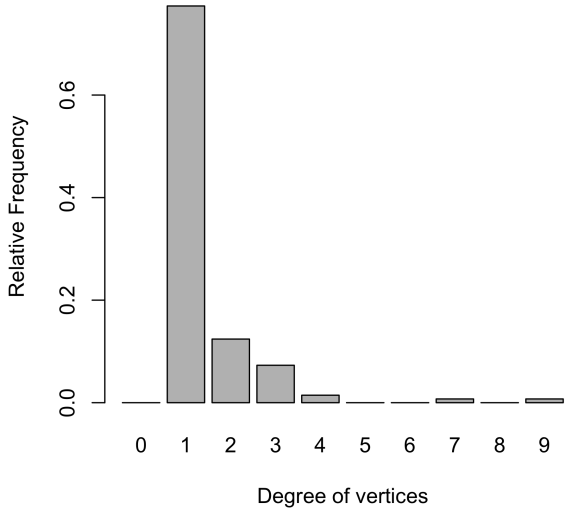
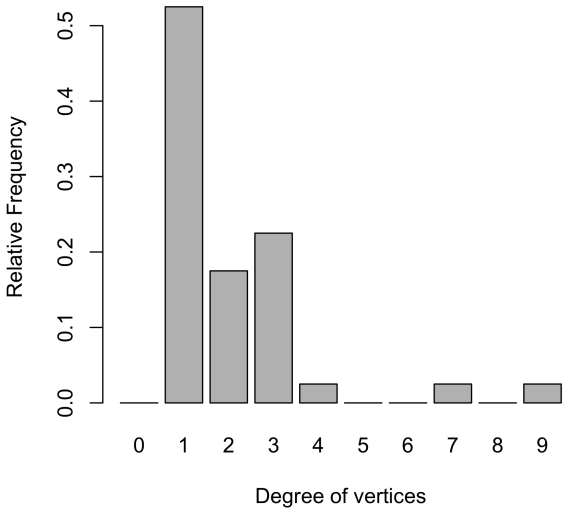


Figure 5: Degree distributions.

(a) Relative frequency of nth degree vertices in our network.



(b) Relative frequency of nth degree vertices in the network's largest component.



A Network theory

The following theory, and more about networks, can be found at Jungnickel's book[1]. A *network* (or *graph* in the mathematics literature) is a pair $\mathcal{N} = (V, E)$ consisting of a non-empty, finite set V and a set E of two-element subsets of V[2]. An element $e = \{a, b\} \in E$ is called an *edge* with *end vertices* a and b. We say that a and b are *incident* with e and that a and b are *adjacent* or *neighbors* of each other, and write $e = ab$. The degree of a vertex v is defined as

$$\deg v := |\{u \in V : u \text{ is adjacent to v }\}|, \tag{1}$$

where $|A|$ denotes the cardinality of a set A. A network is *weighted* if there is a function $w : E \to \mathbb{R}$, and we say and edge e has weight $w(e)$. In a weighted network, we define the strength of a vertex v as the sum of the weights of the edges inciding on v. A sequence $(v_1, \ldots, v_k)$ of adjacent vertices is called a *walk* starting on v_1 and ending in v_k. Two vertices a and b are *connected* if there exists a walk starting in a and ending in b; we say the vertices are *disconnected* if no such walk exists. If all pairs of vertices of a network $\mathcal{N}$ are connected, we say $\mathcal{N}$ itself is connected. Given a network $\mathcal{N} = (V, E)$, and $V' \subseteq V$, we denote by $E_{V'}$ the set of all edges $e \in E$ which have both their end vertices in V'. The network $(V', E_{V'})$ is called the *induced subnetwork* on V'. Each network of the form (V', E') where $V' \subseteq V$ and $E' \subseteq E_{V'}$ is said to be a *subnetwork* of the network $\mathcal{N}$. A *connected component* of a network $\mathcal{N}$ is a connected subnetwork (V', E') such that any vertex in V' is disconnected from vertices not in V'.

References

[1] D. Jungnickel. *Graphs, networks, and algorithms.* Number v. 5 in Algorithms and computation in mathematics. Springer, Berlin ; New York, 1999.

[2] Thomas Kluyver, Benjamin Ragan-Kelley, Fernando Pérez, Brian Granger, Matthias Bussonnier, Jonathan Frederic, Kyle Kelley, Jessica Hamrick, Jason Grout, Sylvain Corlay, et al. Jupyter notebooks—a publishing format for reproducible computational workflows. In *Positioning and Power in Academic Publishing: Players, Agents and Agendas: Proceedings of the 20th International Conference on Electronic Publishing*, page 87. IOS Press, 2016.

[3] Friedrich Nietzsche. *The Antichrist.* Project Gutenberg, September 2006. `http://www.gutenberg.org/files/19322/19322.txt`.

[4] R Core Team. *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria, 2020.

[2] Some literature allows V to be infinite, but it will not be needed in our discussion. Also, for *directed* networks, E consists of ordered pairs of elements of V.