Inspiration for the Project

As soon as I first heard about distant reading, I was reminded of Lotaria, a character from Italo Calvino's novel *If on a winter's night a traveler...* Lotaria is a student who reads books using a machine that counts word frequency. Before I read about Lotaria, I had never considered doing any sort of statistical analysis on literature. In 2016, Lotaria's machine inspired me to write a poem. Her method reminded me of how word clouds visualize word frequencies. I realized that I could manipulate a word cloud by curating the data that I enter into it. By copying and pasting, I created a spreadsheet that had many instances of certain words. The majority of the words are formed by dissecting my name into syllables that have been correctly or incorrectly pronounced by friends and strangers my whole life. I fed the words into an online word cloud generator which produced the following output:



I called this poem "My Modern Lotaria Machine." Knowing more now about the modernist movement, I probably would now question how my work relates to it. Then, I just used it as a synonym for "contemporary."

This week's making exercise gave me a chance to revisit Calvino's ideas. Voyant allows me to read texts using Lotaria's method. Lotaria asks, "What is the reading of a text, in fact, except the recording of certain thematic recurrences, certain insistences of forms and meanings?" (Calvino 186). The assumption is that the word frequencies will reveal the themes of the book, in particular as they relate to her academic theories:

"An electronic reading supplies me with a list of the frequencies, which I have only to glance at to form an idea of the problems the book suggests to my critical study. Naturally, at the highest frequencies the list records countless articles, pronouns, particles, but I don't pay them any attention. I head straight for the words richest in meaning." (186)

Lotaria demonstrates her method in the text by looking at an unnamed novel. From the most frequent words, she determines its genre, and initially critiques that the "narration is entirely on the surface" (187). After looking at the least frequent words, however, which included terms like "underfed" and "underprivileged," she determines that the book "isn't completely superficial, as it seemed" (187).

If on a winter night a traveler... is concerned with how we understand reading, using characters like Lotaria to demonstrate what reading might mean in a certain context. Calvino's writing appears critical of Lotaria's method. Lotaria is a student in a "very important university seminar in literary studies" (185). The narrator, a writer, opines that Lotaria has read his books "only to find in them what she was already convinced of before reading them" (185). Upon looking at "other women reading," he is relieved to "convince [him]self that not all readers are like this Lotaria" (185). His labeling of Lotaria as a "reader" demonstrates an understanding of computational analysis as reading, following a tradition of "reading with and through (and by) the machine" (Johnson and Salter 6).

Interestingly, the narrator ponders how Lotaria's distant reading methods might affect him as a writer. "Now, every time I write a word, I see it spun around by the electronic brain, ranked according to its frequency, next to other words whose identity I cannot know... I try to image what conclusions can be drawn from the fact that I have used this word once or fifty times" (189). Calvino's narrator fears what distant readers might find in his text. "By looking for repeated terms and the phrases in which they are used, we can draw some conclusions about the implicit priorities of the authors of these documents" (Johnson and Salter 3). Calvino implies that these priorities are so implicit that the writer may not even be aware of them. "Perhaps instead of a book, I could write lists of words, in alphabetical order, an avalanche of isolated words which expresses that truth I still do not know, and from which the computer, reversing its program, could construct the book, my book" (189).

The Analysis

Lucky for the narrator, I am here to try that method. For this project, I chose to use distant reading to analyze Walter Isaacson's biography *Elon Musk*, and then use Tracery to generate new text based on the word frequencies. By applying Lotaria's method of

analyzing novels to a biography, I expose the literary nature of historical narratives. It also allowed me to explore Calvino's idea that Lotaria read books only to confirm her own theories. I have both historiographical beef with Isaacson's "great man" history and general beef with the ways Musk has been idolized by business folks and nerds. This would also give me an opportunity to connect to Costanza-Chock's "narratives" principle of Design Justice, which invites the reader to view tech company origin stories through a critical lens (Costanza-Chock 26).

I began by cleaning up the text from the book, which was acquired in an absolutely legal way. I chose to remove the book's front matter and index, leaving only the narrative itself. Already, my data analysis stripped the context of the book away, leaving no mention of the author, the book title, or its date of publication. I realized that I would first need to remove stop words, which Lotaria also ignored. Voyant did not automatically apply common English stop words, so I applied them in the settings. Between these two cleaning steps, a lot of context is ignored. For narratives like these especially it's important to know who is writing and when they are writing. The loss of articles and other stop words remove the connective tissue between words, which could have a large impact on its meaning. I decided to explore this idea further in the writing stage.

Term	Count
the	9036
to	5930
а	5290
and	4606
he	4124
of	3694
was	3029
that	2856
in	2855
musk	2241
it	1992
his	1905
had	1577
with	1516
on	1500
for	1480
at	1269

Following Lotaria's lead, I looked first at the most frequent words. I can speculate a bit about the most common words. Musk's business ventures appear to be a very frequent topic, with "Twitter," "Tesla," and "SpaceX" all appearing in the top ten. The frequent use of "says," "said," and "told" seems to imply that Isaacson's information comes from conversations, presumably with Musk and those who know him. A focus on the technology produced by the companies also seem to be a major focus: "car," "rocket," "engineers," and "launch" appear in the 20s and 30s. Terms like "factory," "design," "engineers" and "team" make me curious about how the book describes labor and the relationships between Musk and his employees. Musk paints himself as a scientist or engineer, and the focus on technology here makes me think that this book may contribute to that mythos.

	Term	Count
1	musk	2241
2	says	626
3	said	620
4	twitter	489
5	like	445
6	tesla	440
7	elon	433
8	just	424
9	musk's	308
10	spacex	306
11	people	292
12	company	289
13	told	286
14	got	282
15	time	280
16	make	279
17	asked	268

18	car					
19	going					
20	new					
21	team					
22	later	223				
23	day	220				
24	rocket	217				
25	went					
26	way					
27	it's					
28	work					
29	kimbal					
30	engineers					
31	called	170				
32	space	167				
33	launch	160				
24	didn't	150				
34	didn't	159				
34 35	didn't decided	159 159				
34 35 36	didn't decided night	159 159 158				
34 35 36 37	didn't decided night years	159 159 158 151				
34 35 36 37 38	didn't decided night years came	159 159 158 151 150				
34 35 36 37 38 39	didn't decided night years came became	159 159 158 151 150 150				
34 35 36 37 38 39 40	didn't decided night years came became want	159 159 158 151 150 150 149				
34 35 36 37 38 39 40 41	didn't decided night years came became want right	159 159 158 151 150 150 149 149				
34 35 36 37 38 39 40 41 42	didn't decided night years came became want right wanted	159 159 158 151 150 150 149 149 148				
34 35 36 37 38 39 40 41 42 43	didn't decided night years came became want right wanted room	159 159 158 151 150 150 149 149 148 147				
34 35 36 37 38 39 40 41 42 43 44	didn't decided night years came became want right wanted room i'm	159 159 158 151 150 150 149 149 148 147 147				
34 35 36 37 38 39 40 41 42 43 44 45	didn't decided night years came became want right wanted room i'm come	159 159 158 151 150 150 149 149 148 147 147 147				
34 35 36 37 38 39 40 41 42 43 44 45 46	didn't decided night years came became want right wanted room i'm come things	159 159 158 151 150 150 149 149 148 147 147 147 144				
34 35 36 37 38 39 40 41 42 43 44 45 46 47	didn't decided night years came became became want right wanted room i'm come things meeting	159 159 158 151 150 150 149 149 149 148 147 147 147 147 144 143				
34 35 36 37 38 39 40 41 42 43 44 45 45 46 47 48	didn't decided night years came became want right wanted room i'm come things meeting year	159 159 158 151 150 150 149 149 149 148 147 147 147 147 144 143 142				



We can also get a sense of the narrative structure. I imagine the narrative is at least somewhat chronological, given the arc of company mentions. His Twitter acquisition is still recent and overshadows the other companies in the later parts of the book.

Like Lotaria, I then sought to add depth to my analysis by looking at the least frequent words. There were too many of these to draw any strong conclusions. I did find some surprising words, like "cowlick," "cowardly," "cocaine," and "barbarians." Perhaps a closer look would find something here, but I can only speculate that the book features many anecdotes that cover a wide variety of topics. Roughly half of the words in the corpus were only used once, which seems like a high variety to me, but I do not have a good sense of how word frequencies are typically distributed.

The Visualization

For my visualization, I decided to reconstruct the book from its individual words. Like the grammar last week, this is a form of what Leonardo Flores calls "distant writing." Flores presents a talk that "explores the work of writers of electronic literature who, instead of writing sequences of words directly, create a computer program or modify an existing one to generate their intended texts" (Flores). Calvino describes taking a list of word frequencies and generating a text based on that. I did something similar with "My Modern Lotaria Machine." Here, I chose to add the stop words back in, because articles and conjunctions seem important to understanding the meaning. They also have the

ability to shift the meaning of sentences. A "not" or a "but" can invert the apparent valence of a phrase. I exported the terms and frequencies from Voyant as a tab-separated .txt file.

Term	inDocumentsCount			RawFrequency		RelativeFrequency	
the	1	9036	0.04857	8817	null	null	0.048578817
to	1	5930	0.03188	052	null	null	0.03188052
a	1	5290	0.02843	979	null	null	0.02843979
and	1	4606	0.02476	251	null	null	0.02476251
he	1	4124	0.02217	1209	null	null	0.022171209
of	1	3694	0.01985	9469	null	null	0.019859469
was	1	3029	0.01628	4334	null	null	0.016284334
that	1	2856	0.01535	4261	null	null	0.015354261
in	1	2855	0.01534	8884	null	null	0.015348884
musk	1	2241	0.01204	7933	null	null	0.012047933
it	1	1992	0.01070	9275	null	null	0.010709275
his	1	1905	0.01024	155	null	null	0.01024155
had	1	1577	0.00847	8176	null	null	0.008478176
with	1	1516	0.00815	0231	null	null	0.008150231
on	1	1500	0.00806	4212	null	null	0.008064212
for	1	1480	0.00795	669	null	null	0.00795669
-+	4	1000	0 00000	2224		11	0.000000000

I had big ideas for this project. I wanted to use the word frequencies to generate another body of text. I thought that generating another text that has the exact same frequency of words would demonstrate the limits of simple word frequency analysis – if two texts with very different meanings could have the same distribution, how useful can word frequency analysis be? I wanted the output to be intelligible, so I had a whole plan about sorting the words into nouns, verbs, etc. and then using Tracery to construct sentences that pull from the frequencies. I couldn't figure out a good way to break up the word types, though, and I realized that I would have to come up with like a billion sentence structures. So I decided to instead just randomize the words regardless of their word forms.Based on how probability works in Tracery, I needed to expand the list so that each word was listed the number of times it appears. So instead of "the" and "9036," I needed a list that included 9036 individual instances of the word "the". I figured I could do this in Excel? Using the wisdom of stackoverflow user bosco_yip, I had it working in Excel. Except I had to manually drag the formula down.

I got about 137,000 words in before I realized that I was just sorting all of the words in the book into alphabetical order, with more steps. I finally decided to just take the novel text and feed it into a word randomizer called<u>"word shuffler"</u>. The way the word shuffler tokenized its words had the benefit(?) of keeping the punctuation. I was worried that a random string of words would just be completely unreadable, but this at least gives the appearance of readability...? While the output is mostly nonsense, it did generate a few interesting lines. Since "Musk" featured so prominently in the data set, I expected to find some interesting juxtapositions.

"public Musk good" - I do not think of public goods when I think of Musk

"Musk reality-bending pronouns" – oh boy

"mind-bogglingly Musk was Dragon" – this appears particularly sycophantic.

Ultimately, did I even create a visualization? I am reminded of the epigraph Benjamin Peters used from Alfred Korzybski – "The map is not the territory," from *Your Computer is on Fire* (Peters 71). A data visualization is typically a layer of abstraction away from the original text. There is still a difference between the original text and my randomized "visualization," of course, but it also brings to mind Jorge Luis Borges' story "On Exactitude in Science," which imagines a map that is so accurate that it is the size of the territory it represents. My visualization is exactly as long as the original book. It seems that the benefits of data visualizations, like condensing a large amount of information into a small readable format, was not captured well at all. I thought the prevalence of certain words might become more apparent in the generated version, but the sheer scale of the text drowns them out. I will have to think more about this kind of recursive distant writing, and what it can tell us about the original text and the methods used to generate the new one.