List Order Grant Proposal

Yiu-On Li · Dec 8, 2022 · ENGL 197

Abstract

We seek $50,000 to investigate how people prefer to order lists, such as “he and she” as opposed to “she and he,” across as many mediums as possible, including novels, social media posts, and news articles. We hope to use this data to identify hidden biases in thinking and behavior in both humans and large language models alike, so that we may correct for them in both cases.

After scraping and cleaning data from the web, we would need to isolate the lists in that data. What constitutes a list varies from source to source, but for this study we restrict the definition of a list to a collection of words separated by commas and/or conjunctions. To isolate such lists, we would use AntConc or a more optimized alternative to search for telltale conjunctions and punctuation, with the goal being to eventually train a language model to automatically isolate lists for us.

Once we have our lists, we would then topic model them so that we can cross-reference topics with their constituent lists for insight on what people value. Finally, we would compare the lists we have gathered with those generated by large language models like ChatGPT to see if and how bias is present and deviates.

Narrative

What is the difference between someone who says “they, he, and she”; someone who says “she, he, and they”; and someone who says “he, she, and they”? Is it “Ghanian, Canadian, Indian, and American,” or is it “American, Canadian, Indian, and Ghanian”? And who is the sort of person that puts carrots before milk, and milk before carrots, on their shopping list?

More specifically, how does the order in which people structure various lists of words and items reveal certain values and hidden biases?

To answer these sorts of questions, we seek a grant of $50,000.

Such studies of bias have gained greater urgency with the rapid advancement of large language models and natural language processing, tools that devour petabytes of human-generated text and, in response to text prompts, generate even more text based on the patterns they detect in their massive samples. Because humans are biased, the models we create—by feeding them biased material from, say, the Internet—likewise have the potential for bias. Examples of bias range from subtly exclusionary language, like “women doctors,” to hate speech (Bender et al. 613).

But if we can detect, weed out, and correct for bias at the point of generation—or, even better, if we can detect, weed out, and correct for bias at the data curation or training stage—then the models we create will be that much more representative of all of humanity, and will therefore service that much more of humanity (Bender et al. 615).

Detecting that bias will require an understanding of how bias manifests. And list order is one of many avenues of investigation.

It is possible that someone who tends to list “he” before “she,” or lists “John” before “Jane,” unconsciously expresses a patriarchal bias. And it is possible that someone who tends to list a certain nationality or religion before others places greater importance and primacy on that nationality or religion.

It is also possible that list order reveals nothing of value systems. Perhaps list order is random, or perhaps it is unique to everyone. Perhaps “carrots and milk” is no different than “milk and carrots.”

Both possibilities have the potential to uncover aspects of the human psyche. If it is the first possibility, then we will have taken steps toward discerning—and eventually dispelling—those manifestations of bias in human and machine alike. As Fred Rogers, host of *Mister Rogers' Neighborhood*, once said about feelings—and, by extension, biases—"Anything that is mentionable can be more manageable" (Farmer). If it is the second possibility, then we will have gathered more support for the position that the mind is indeterminate in regards to its listmaking, and we can divert more of our attention to other areas of study in bias and discourse (or is it discourse and bias?).

For the purposes of this experiment, we restrict the definition of a list to a collection of words separated by commas and/or conjunctions—or, more technically, comma-separated values. As such, lists not specifically about words—for instance, Gass gives one possible definition for a film as "a series of shots in the order of their snapping"—would not count as a list in this study (28).

Large lists or writings where one could make an argument for exhibiting listlike qualities—such as top tens, census data, and the order of the sentences and paragraphs of a paper—also exceed the scope of this study. These collections begin to exhibit qualities of databases as opposed to streamlined lists, and at this point we start to lose the spontaneity that characterizes natural human discourse.

In other words, we are looking merely at lists whose units comprise one word—or at most a handful of words—at a time. It is in these relatively simple lists, created without much conscious effort, that may best reveal the creator's subconscious tendencies.

After exploring which list orders people prefer (if any), we hope to use that data to construct the "typical" human. For example, if most people put "she," "Buddhism," and "bread" as the first unit of their lists, then we would define the typical human to be a Buddhist who likes bread and uses she/her pronouns. We would then like to compare the accuracy of this result to actual data on the proportion of Buddhists in the world as opposed to people who practice other religions, worldwide bread consumption as opposed to other foods, and so on.

In the final stage of our study, we intend to use existing large language models, such as ChatGPT, GPT-3, and BERT, to generate a large volume of lists based on topics discussed previously (such as pronouns, nationality, and groceries). We would then compare these generated lists to the lists we have gathered on our own to see whether and how they deviate. If manual comparison proves to be too slow, cumbersome, and subjective, it may become necessary for us to train our own language model to compare many different pairs of lists and generate something akin to a similarity index for each topic.

Once we have determined the degree of similarity between human and machine for certain topics—and, by extension, the degree of difference—we can then take steps to correct for that difference. Not just to move machine closer to human, but to move machine beyond human: a machine more capable of neutrality than humans will ever be. In the process, perhaps humans will be able to learn from machines and become more neutral themselves.

Environmental Scan

Few studies regarding list order seem to have been conducted prior.

The closest would seem to be an article by Gass in the magazine *Salmagundi*. Gass takes an exhaustive look at lists, their components, their uses, their purposes, and their contexts. Most importantly for this study, Gass lists several "organizing principles" for lists:

- spontaneity (as in order of observation),
- hierarchy (as in alphabetically),
- chronologically or simply logically (as in a table of contents), and
- "some principle of value or importance" (23).

That last principle is the object of this present study, for Gass makes little elaboration on what those principles or values might be. And, as a whole, the article does not give much empirical evidence, relying mostly on the author's own observations.

Next is the book *Sorting Things Out* by Bowker and Star. The book goes into great detail on classification theory and the different categories that pervade modern society, from politics to medicine to race. And where there are categories, there are lists containing the members of those categories.

As for the social and moral implications of categories, the authors observe that "a classification is not of itself an explanation," but, "perceived as real, it has real effect" in helping to "tie the person into an infrastructure." They cite an *X-Files* episode where, upon a murderer being labeled a “homicidal maniac,” he gains the conviction of one (Bowker and Star 319).

Unfortunately, the book discusses only categories and their members. The book does not discuss order and how order might change the meaning of a category—or if order might even give rise to new categories altogether.

A final study, by Kan and Ross, applies to computing: It investigates how to order a list such that “the long-run average cost” of retrieving an element from said list is minimized (1004). This method of ordering a list largely eschews the content of the element in favor of its accessibility—similar to social network theory, which emphasizes the edges connected to a node and its position within the network over the identity of the node itself when it comes to its “opportunities and constraints” (Borgatti et al. 894).

While optimizing for efficiency of retrieval is certainly one way to order a list, such a method requires deliberate intent and may not be top of mind for humans casually writing down a series of words. In any case, the article discusses ideal list formation, and not formation as is typically done by people, biases and quirks and all.

Materials, Methods, and Outputs

Deciding on the Data

The first and most crucial step in this study is to gather lists. Lots and lots of lists. (Refer to the Narrative section for what constitutes a list in this study.)

Any source will do. Perhaps there is a marked difference in list order between writing and speaking, but for now we group them together. And perhaps we ought to favor more contemporary writing if we wish to capture contemporary styles and biases—but again, for now we seek old and new alike.

The goal is to amass as many lists from as many domains as practical. Novels, screenplays, social media posts, speeches, news articles—the list goes on. We do not rule out linguistic corpora at this stage, but such a source would require extensive cleaning to remove timestamps and linguistic symbols, and even then the constant interplay between and interruption of different speakers may make it difficult to isolate individual lists for analysis.

Much of this data can be scraped from the web. Social media posts—Twitter in particular—will require specialized tools, such as Netlytic or SocioViz, to acquire.

Gathering the Text

For this proposal, we will demonstrate the methods of our study by analyzing more than a thousand texts from the English collection of Project Gutenberg. Various methods exist for acquiring this collection, of which we try several: installing GNU Wget to retrieve files via the Hypertext Transfer Protocol, downloading web reader Kiwix and its accompanying ZIM file of English Gutenberg content, and writing scripts with the aid of Python packages and various GitHub repositories (“Download/en”, “GNU Wget”, Kiwix”, “Main Page”, “Offline Catalogs”, puntonim and Gleave, “Robot Access to Pages”, Simončič).

The most promising of these approaches lies in the Python packages and GitHub repositories—they allow straightforward bulk download of plain text files, after which we can manipulate and dissect them as we please. It is at this point we switch from working in our native Windows environment to an Ubuntu Linux distribution via the Windows Subsystem for Linux, as Linux enjoys greater support and documentation for package installations (“Install WSL”).

The specific Python package we use is the aptly named Gutenberg, whose features include both bulk download and text cleanup of headers and footers. To install, we run the following terminal commands (Wolff):

```
$ sudo apt-get install libdb++-dev
$ pip install gutenberg
```

We then write a script to download, clean, and save a little over a thousand texts from Project Gutenberg:

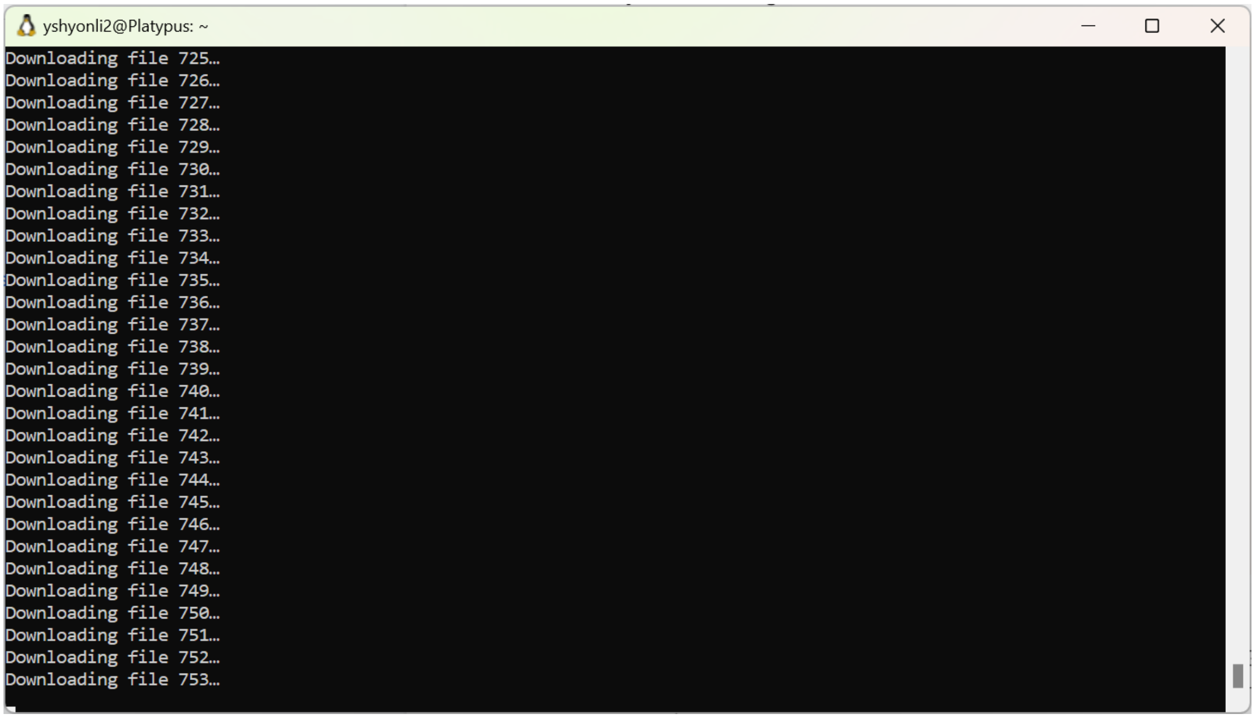
```
from gutenberg.acquire import load_etext
from gutenberg.cleanup import strip_headers
from gutenberg._domain_model.exceptions import UnknownDownloadUriException

directory = "gutenberg-dump/new-txt"

for i in range(1, 1100):
  try:
    print(f"Downloading file {i}…")
    text = load_etext(i)
    file = open(f"{directory}/{i}.txt", 'x')
    textCleaned = strip_headers(text).strip()
    file.write(textCleaned)
    file.close()
  except UnknownDownloadUriException:
    print(f"File {i} not available")
    continue
  except FileExistsError:
    continue
```

We skip downloading a file in two cases:

1. When, for whatever reason, a file is not available from Project Gutenberg.
2. When we have already downloaded a file (in case of a crash and we have to rerun the script).

We then run the script with the following command:

```
$ python3 downloadGutenberg.py
```



The final count of texts in our proof-of-concept corpus stands at 1,071.

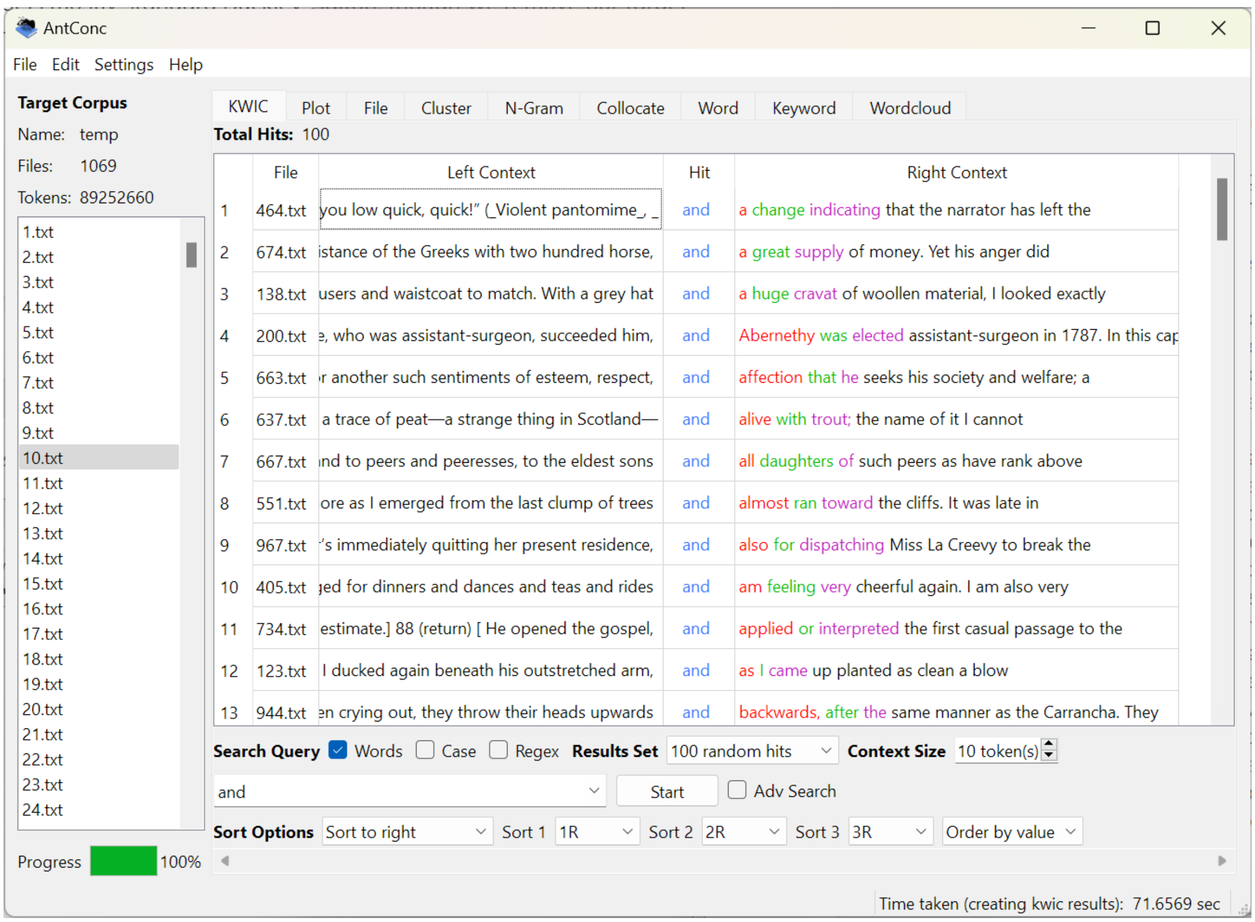
Isolating the Lists

The next step is to comb through each of our 1,071 texts and isolate their lists. Anthony's AntConc will be our tool of choice here.

Our search process mixes automatic detection with manual vetting. Specifically, we will search for the conjunctions "and" and "or," as they tend to be one of the more common building blocks of lists, and then use AntConc's Key Word in Context feature (KWIC) to see which of those words are truly part of a list. We then copy the full list to a separate text file.

Our stopwords list is mostly standard Buckley-Salton, though we remove our target conjunctions ("and" and "or") and pronouns ("he," "she," "they," and "it") for better listmaking.

This is what the search for "and" looks like in AntConc, showing 100 random hits:



And from those 100 hits, we get 73 lists, of which 20 are shown here:

```
two hundred horse, and a great supply of money
trousers and waistcoat
a grey hat and a huge cravat of woollen material
esteem, respect, and affection
society and welfare
peers and peeresses
the eldest sons and all daughters of such peers
dinners and dances and teas and rides
applied or interpreted
upwards and backwards
trees and houses and churches
Haarlem and Amsterdam
slide and carry you down and bury you
great size and strength
support and consideration
bailiffs and constables
rapidly and continually
International Bank for Reconstruction and Development (IBRD), International
Development Association (IDA), and International Finance
the trend of conversation and direction of glasses
ballot-boxes, suffrages, French Revolutions
```

There are, of course, issues with this half-automatic, half-manual approach:

1. Many lists remain unextracted, even under the relatively narrow definition we established in the Narrative section. For example, lists without a conjunction will not show up at all unless they happen to be noticed adjacent to a list with one, and lists that use less common but still relatively frequent separation schemes—such as semicolons and periods—may be underrepresented.
2. Although AntConc helps us considerably in reducing the amount of text we have to sort through, manually verifying and recording lists is still time-consuming.
3. At slightly above a thousand texts, lag is already noticeably present in AntConc. This is not a tool that will scale to the many more thousands and perhaps millions of texts we wish to examine.

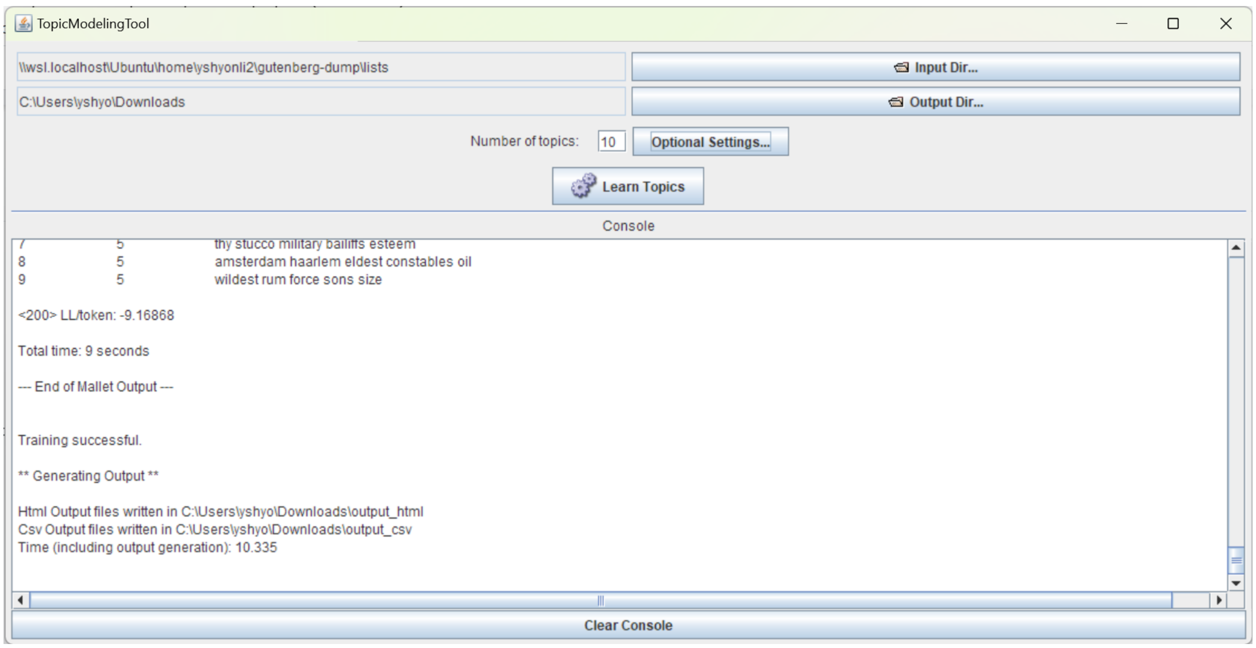
If our proposal is approved, we hope to dedicate more time to finding new and quicker ways to isolate lists. Possibilities include tokenizing our corpus with the Python Natural Language Toolkit and training a language model to recognize lists for us.

Finding Patterns from Lists

In any case, we have some lists now. What do we do with them?

Well, we could do worse than finding out what people usually write about in lists. In other words, we need topic modeling.

We use Enderle's Topic Modeling Tool to find the top topics of the 73 lists we isolated earlier. We exclude stopwords, because we already filtered out stopwords in AntConc. Here is the tool in action:



Here are the 10 topics we get out of the model, training on 200 iterations (because any more and the tool crashes from not enough data):

```
1. esteem olive stripped cheek society
2. loudest sour strength smiled slow
3. angels strong clamor hat grey
4. international trumpets bailiffs affection oil
5. development flower taste haarlem telegraphs
6. peers stucco whiskers amsterdam constables
7. rum circumstances force daughters cravat
8. thy wildest moment woollen fine
9. backwards upwards popular sons eldest
10.   destructive wreaths blinds smallpox military
```

Our dataset is quite small at this stage, so the insights we can draw from these topics are limited. However, we notice that topic 1 perhaps has to do with high society, topic 5 with technology and advancement, and topic 10 with destruction.

While a weakness of topic modeling in the context of our study is that it gives no indication of order, we can cross-reference topics with the original lists we isolated to see if we can discern more patterns. For example, the Topic Modeling Tool, for topic 1, gives as its primary members the lists “my cabinet, and the two chests of gold, and the tablets” and “clanking armor and lance and shield.” Given the primacy of “my cabinet” and “clanking armor,” one begins to wonder whether high society favors a heavy and substantial presence of self. Meanwhile, topic 10 has as its primary lists “secure and publish”; “indiscreet conduct and want of tact”; and “International Bank for Reconstruction and Development (IBRD), International Development Association (IDA), and International Finance.” Perhaps people favor security and solidity of behavior as a guard against calamity.

At present, we cannot say for sure—we need a lot more lists and a lot more topics. But if our proposal is approved, we would have the opportunity to get those lists and topics, and eventually we may be able to construct a profile—or several profiles—of what a “typical” human being believes and acts like.

Comparing Lists with Large Language Models

Speaking of behavior, how do our mechanical counterparts behave compared to us? The final part of our study would see us taking our lists and topics and putting them against the lists and topics that a large language model would generate. We intend to explore if these models have hidden preferences and, if so, what these preferences look like, despite their claims to the contrary.

Are you biased?

As a machine learning model, I am not capable of forming opinions or beliefs, and therefore cannot be biased. I am programmed to provide accurate and helpful information to the best of my abilities based on the input I receive. I am designed to be as neutral and impartial as possible.

The previous snippet comes from ChatGPT, a recently released large language model that's widely considered to be on the cutting edge of generative text, at least when it comes to what the public has access to (Lopez).

To minimize the human bias that is imprinted on the model at the point of generation (such as by prompting it with "generate a list like 'he, she, and they'"), we will simply prompt the model with a topic. For example, based on the list "esteem, respect, and affection" from this proposal's demonstration corpus, our prompt to ChatGPT is "list of personal values." Here are the results:

```
1. Respect
2. Honesty
3. Loyalty
4. Fairness
5. Responsibility
6. Compassion
7. Integrity
8. Courage
9. Perseverance
10.   Open-mindedness
11.   Empathy
12.   Self-improvement
13.   Creativity
14.   Independence
15.   Sincerity
```

```
16.   Gratitude
17.   Humility
18.   Positive attitude
19.   Forgiveness
20.   Love
```

The model repeats “respect” and bumps it from a secondary position to the top spot. Might this reveal a favor for the quality of respect over the original list’s preference for a more self-centered “esteem”? Though, like the original list’s “affection,” ChatGPT places “love” at dead last.

For the list “dinners and dances and teas and rides,” we prompt ChatGPT with “list of social pastimes.” Here is what it gives:

```
1. Going to parties or social events
2. Spending time with friends and family
3. Playing sports or other group activities
4. Attending concerts or other live performances
5. Going to the movies or theater
6. Participating in community service or volunteering
7. Going to the park or beach
8. Attending religious or spiritual gatherings
9. Playing board games or card games with others
10.   Going to the gym or joining a fitness class
11.   Going on outings or adventures with friends
12.   Cooking or baking together
13.   Visiting museums or art galleries
14.   Going to the library or bookstore
15.   Attending classes or workshops to learn new skills or hobbies.
```

When considering only the first item from both lists, both lists are about equal in terms of favoring an activity where multiple people participate. However, the original list’s “dinner” has an element of gluttony and self-indulgence that ChatGPT’s “going to parties or social events” does not.

Overall, with more data, we expect to find little to no difference between human preferences for ordering lists and large language model preferences. Language models are, after all, trained by us with material from us. If nothing else, they will be an uncanny mirror that reflects our own painful insufficiencies back at us.

Works Cited

Anthony, Laurence. "AntConc." *Laurence Anthony's Website*, Laurence Anthony, 1 December 2022, https://www.laurenceanthony.net/software/antconc/.

Bender, Emily M., et al. "On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? 🦜." *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, Association for Computing Machinery, 2021, pp. 610–623, doi:10.1145/3442188.3445922.

Borgatti, Stephen P., et al. 'Network Analysis in the Social Sciences'. *Science*, vol. 323, no. 5916, 2009, pp. 892–895, doi:10.1126/science.1165821.

Bowker, Geoffrey C., and Susan Leigh Star. *Sorting Things Out: Classification and Its Consequences*. MIT Press, 1999.

"Convert PDF to TXT file using Python." *AskPython*, AskPython, https://www.askpython.com/python/examples/convert-pdf-to-txt.

Enderle, Jonathan Scott. "Quickstart Guide." *Topic Modeling Tool Blog*, 6 January 2017, https://senderle.github.io/topic-modeling-tool/documentation/2017/01/06/quickstart.html.

Farmer, Deborah. "What's mentionable is manageable: Why parents should help children name their fears." *The Washington Post*, 20 September 2018, https://www.washingtonpost.com/news/parenting/wp/2018/09/20/whats-mentionable-is-manageable-why-parents-should-help-kids-name-their-fears/.

Gass, William H. "I've Got a Little List." *Salmagundi*, no. 109/110, 1996, pp. 20–38. *JSTOR*, http://www.jstor.org/stable/40548863.

"GNU Wget." *GNU Operating System*, Free Software Foundation, https://www.gnu.org/software/wget/.

"Install WSL." *Microsoft Learn*, Microsoft, 22 November 2022, https://learn.microsoft.com/en-us/windows/wsl/install.

Kan, Y. C., and S. M. Ross. "Optimal List Order under Partial Memory Constraints." *Journal of Applied Probability*, vol. 17, no. 4, 1980, pp. 1004–15. *JSTOR*, https://doi.org/10.2307/3213210.

Kiwix. “Gutenberg Offline.” *GitHub*, GitHub, https://github.com/openzim/gutenberg.

Lopez, German. “ChatGPT is Social Media’s Newest Star.” The New York Times, The *New York Times Company*, 8 December 2022, https://www.nytimes.com/2022/12/08/briefing/ai-chatgpt-openai.html.

“Main Page.” *Kiwix*, 12 June 2022, https://wiki.kiwix.org/wiki/Main_Page.

“Natural Language Toolkit.” *Natural Language Toolkit*, NLTK Project, https://www.nltk.org/.

“Netlytic.” *Netlytic*, Netlytic.org, https://netlytic.org/home/.

“Offline Catalogs.” *Project Gutenberg*, https://www.gutenberg.org/ebooks/offline_catalogs.html.

puntonim, and Adam Gleave. “GUTENBERG-BULK-DOWNLOADER.” *GitHub*, GitHub, https://github.com/puntonim/gutenberg-bulk-downloader.

“Robot Access to Pages.” *Project Gutenberg*, https://www.gutenberg.org/policy/robot_access.html.

Simončič, Jernej. “Windows binaries of GNU Wget.” *eternallybored.org*, https://eternallybored.org/misc/wget/.

“SocioViz.” SocioViz, https://socioviz.net/.

Wolff, Clemens. “Gutenberg.” *GitHub*, GitHub, https://github.com/c-w/gutenberg.