

Applications of GPT in Political Science Research: Extracting Information from Unstructured Text

Kyuwon Lee*, Simone Paci†, Jeongmin Park‡, Hye Young You§, Sylvan Zheng¶

Abstract

This paper explores the use of large language models (LLMs), specifically GPT, for enhancing information extraction from unstructured text in political science research. By automating the retrieval of explicit details from sources such as historical documents, meeting minutes, news articles, and unstructured search results, GPT significantly reduces the time and resources required for data collection. The study highlights how GPT complements human research assistants, combining automated efficiency with human oversight to improve the reliability and depth of research. This integration not only makes comprehensive data collection more accessible, but also increases the overall research efficiency and scope of research. The paper highlights GPT's unique capabilities in information extraction and its potential to advance empirical research in the field. Additionally, we discuss ethical concerns related to student employment, privacy, bias, and environmental impact associated with the use of LLMs.

*Department of Political Science and International Relations, University of Southern California, kyuwonl@usc.edu

†Department of Political Science, Stanford University, simone.paci@stanford.edu

‡Department of Politics, Princeton University, jp49@princeton.edu

§Department of Politics and Public Affairs, Princeton University, hyou@princeton.edu

¶Department of Politics, New York University, saz310@nyu.edu

Introduction

In the expanding landscape of political science research, the integration of advanced artificial intelligence tools has opened novel avenues for data collection, annotation, and analysis. Among these tools, large language models (LLMs), such as OpenAI’s Generative Pre-trained Transformer (GPT), have garnered attention for their potential to enhance research productivity and expand empirical research capabilities (Ziems et al. 2024).¹ This paper specifically examines the use of GPT for information extraction from unstructured text—an essential task that involves retrieving explicitly stated details that may be challenging to access manually. Unlike broader applications such as generating text labels for classification (Wang 2023; Chiu, Collins, and Alexander 2022), simulating survey responses (Argyle et al. 2023b), generating stimulus for survey experiments (Velez and Liu 2024), or engaging in conversations with humans (Argyle et al. 2023a), information extraction focuses on accurately identifying and retrieving explicit content within documents. While GPT shows promise in various tasks, this paper highlights their particular effectiveness in information extraction.

Our study is divided into detailed examinations of the utility of GPT for various data collection tasks. In these examples, GPT’s applications demonstrate its versatility in handling increasingly complex information tasks across two languages: English and Italian. In the first example, GPT is used to clean Optical Character Recognition (OCR) errors from scans of historical documents, demonstrating its basic ability to process textual data. Moving on to more complex applications, in the second and third examples, GPT helps to extract participant information from semi-structured administrative meeting minutes data and detailed

1. The terms “LLMs” and “GPT” are related but refer to different concepts in the field of natural language processing and artificial intelligence. LLMs are a category of natural language models trained on massive datasets containing diverse language data and are designed to produce coherent human-like text by predicting the next word in a sequence. The GPT family of LLMs refer to a specific set of models developed by the company OpenAI. These models were among the first to combine unprecedentedly large training sets (over 570gb of text for GPT-3) with the transformer architecture, a neural network design that allows the model to pay varying amounts of attention to different parts of the text, allowing it to more effectively understand context and relationships within text. In summary, GPT is a specific implementation of the broader category of LLMs.

source information from lengthy news articles. In the final example, we show GPT’s ability to perform an advanced task of synthesizing data from multiple internet sources.

Each of these applications demonstrates how GPT performs labor-intensive tasks not only with remarkable speed but also with accuracy that either matches or exceeds human efforts. Furthermore, the use of GPT in these contexts highlights its potential to handle large volumes of data, a capability that is particularly useful in political science where researchers are often faced with extensive but only partially structured datasets. The examples we present in this article highlight GPT’s strengths in natural language processing while mitigating its weaknesses in complex reasoning and hallucination (i.e., false information) (Ji et al. 2023; Wei et al. 2022) along with the reliability and consistency of synthetic survey data produced by LLMs (Bisbee et al., Forthcoming).

By presenting a range of unique examples, this paper aims to expand thinking in the discipline about the potential uses of LLMs rather than provide a specific how-to guide. We discuss the importance of creatively engineering prompts tailored to different tasks, illustrating that the first prompt might not always suffice, and that careful refinement is crucial for optimal results. Through this approach, we seek to inspire further exploration and creative problem-solving using LLMs in political science research.

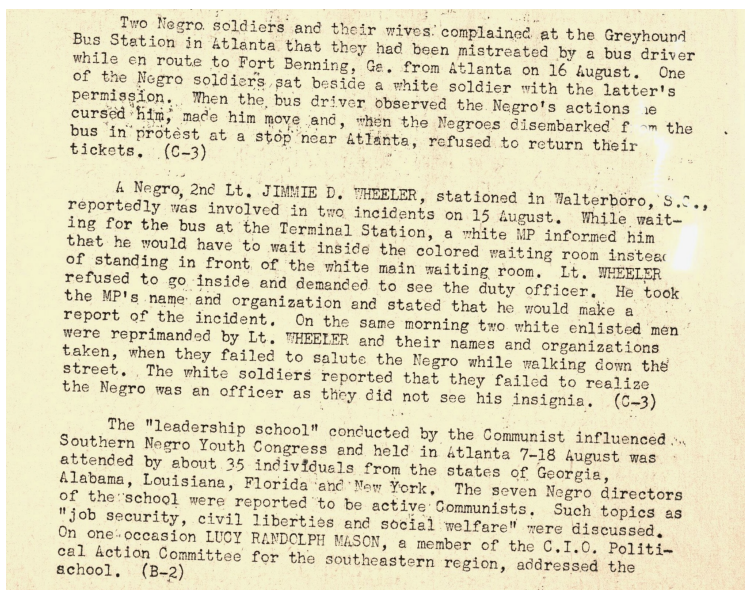
GPT’s potential to reduce the gap in unequal research resources is another significant benefit of its inclusion in the political science toolbox. Traditionally, large-scale research projects often have been the purview of well-funded researchers who can afford large teams of research assistants (RAs) and expensive data processing tools. However, GPT’s ability to automate and streamline data extraction and analysis tasks could level the playing field, allowing researchers with limited budgets to undertake more extensive research efforts. At the same time, the use of LLMs in research raises ethical concerns, including the potential loss of jobs for student research assistants, privacy risks, social bias in outputs, and significant environmental impacts. We will discuss the various ethical concerns of using GPT in detail.

Applications

Example 1: Cleaning and Analyzing Historical Data

This section explores the use of GPT in conjunction with OCR tools to clean and analyze historical documents. While OCR technology has advanced, the quality of OCR output still depends on the quality of the scanned image and the choice of OCR tool, which often results in errors, such as misspellings and odd spacing. High-quality OCR tools such as Google Cloud Vision (GCV) produce cleaner text, but are often impractical due to issues such as document accessibility and other resource constraints. To address these challenges, we use GPT to clean text produced by the open-source OCR tool, Tesseract.

Figure 1: Example of a Scanned Image from a Weekly Intelligence Report



We employ previously unused archival materials concerning World War II-era race-related incidents and racial reform from the National Archives at College Park, Maryland (NAID 615574). These materials, consisting of five boxes, contain the weekly intelligence reports of the Army Service Forces from August 1944 to January 1946. The reports provide a comprehensive description of race-related incidents involving military personnel, in addition

to the preventive or reactionary measures taken to mitigate racial strife (sample image in Figure 1). The records contain important details about these incidents, such as the date and location, the people involved, and the actions taken by key players. Unfortunately, available OCR tools show varying levels of accuracy (Table A1).

Table 1: OCR Results Using Tesseract and GPT

Description	Result
Prompt	"Do not change the content of the text itself, just take out noise. Also, when you see parentheses with uppercase alphabet and number inside, like (A-2), just leave it:"
OCR (Tesseract)	Two Negro. soldiers and their wives? complained. at the Greyhound ay ea . Bus Station in Atlanta that they had been mistreated by a bus driver i a F while en route to Fort Benning, Ge. from Atlanta on 16 4ugust . One ee: Be } of the Negro soldiers, sat beside a white soldier with the latter's [fo a cS permission, When the bus driver observed the. Negro'ts actions 16 as Ls. curséd him } made him move .and, whén the Negroes disembarked f -m the ' . fo Bae be bus in'protést at a stop near Atlanta, refused to return their i ie a tickets, _(C-3)
Cleaned with GPT	Two Negro soldiers and their wives complained at the Greyhound Bus Station in Atlanta that they had been mistreated by a bus driver while en route to Fort Benning, Ga. from Atlanta on 16 August. One of the Negro soldiers sat beside a white soldier with the latter's permission. When the bus driver observed the Negro's actions cursed him, made him move and, when the Negroes disembarked from the bus in protest at a stop near Atlanta, refused to return their tickets (C-3).
GCV	Two Negro soldiers and their wives complained at the Greyhound Bus Station in Atlanta that they had been mistreated by a bus driver while en route to Fort Benning, Ga. from Atlanta on 16 August. One of the Negro soldiers sat beside a white soldier with the latter's permission. When the bus driver observed the Negro's actions e cursed him, made him move and, when the Negroes disembarked from the bus in protest at a stop near Atlanta, refused to return their tickets. (C-3)

* The errors are highlighted in bold text.

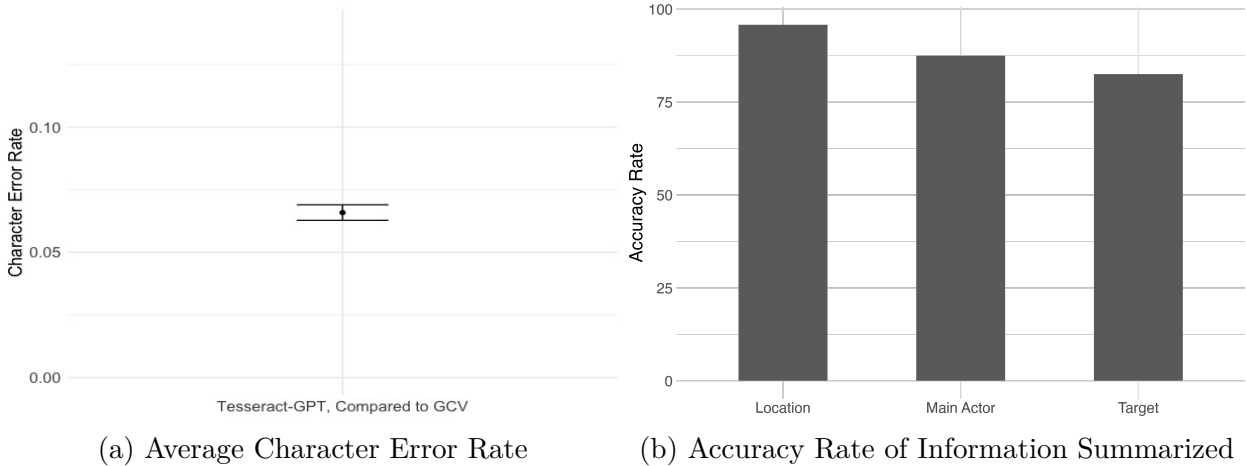
Here, we propose a time-saving approach that combines open-source tools (**Tesseract**) with GPT. We take the noisy text generated by **Tesseract** and use the GPT API to clean the noise, a process illustrated in Table 1. We then visualize the performance of this method compared to **GCV**-processed text for an entire box, consisting of 20 folders (997 images).² We use the **GCV**-processed text as gold standard data because of its superior quality once images are obtained and pre-processed for accurate character recognition, in addition to the impracticality of generating human-typed gold standard data for large archival materials. We measure the performance of the **Tesseract**-GPT combination using Character Error Rate (CER), a common metric used to evaluate OCR performance. CER is defined as the ratio of the number of character-level errors to the total number of characters in the reference text. Additionally, we use GPT to extract critical details from each incident, including the location, main actors involved, and the targets. Finally, we extract a 10% sample from the cases, and manually verify the accuracy of the information extracted by GPT.

The results shown in Figure 2 illustrate the effectiveness of GPT in cleaning and analyzing historical data. On average, about 6% of the characters in the OCR-generated text are incorrect, compared to the text generated by **GCV**, our reference text (Figure 2a). Furthermore, the high accuracy rate for capturing relevant information—such as location, main actor, and target—illustrates the general effectiveness of GPT in information extraction, especially when it comes to objective, context-independent information, such as location (Figure 2b).³

2. Box 262, Security Classified Reports and Memorandums Concerning Race Relations in the United States and Overseas, August 1944–January 1946. Records of the Office of the Secretary of War, Record Group 107. National Archives Building, Washington, DC.

3. Most errors regarding the main actor and target identification are due to contextual factors. GPT has difficulty identifying the target when it is not explicitly mentioned, such as in nonviolent cases, or when the main actor does not have an explicit target (e.g., NAACP meetings). It also has difficulty identifying the main actor in indirect descriptions (e.g., news articles).

Figure 2: Performance of GPT in Cleaning and Analyzing Archival Data



Example 2: Extracting Unstructured Administrative Data

In this section, we show how GPT can be used to collect and clean administrative data provided in a semi-structured format (often in PDFs). We focus on meeting minutes from federal advisory committees (FACs) within federal agencies in the United States. A significant number of FACs serve as independent advisors that make policy recommendations to federal agencies. These committees hold over 5,000 public meetings annually, bringing together committee members, federal agency officials, and outside interest groups to discuss agency policy. As such, FAC meeting minutes provide a unique opportunity for scholars to examine the extent to which outside groups participate in bureaucratic policymaking. Figure 3 shows two examples of committee meeting minutes, one from the Environmental Protection Agency (EPA) and one from the Centers for Disease Control and Prevention (CDC). Each includes the names, positions (e.g., chair, members, agency staff, or public attendees), and affiliations (e.g., Karmanos Cancer Institute) of meeting participants.

Our goal is to extract the participant information from 79 meeting minutes of two Environmental Protection Agency (EPA) FACs: the Clean Air Scientific Advisory Committee (CASAC) and the Science Advisory Boards (SAB), from 2017 to 2023. We use the GPT API

Figure 3: Examples of Advisory Committee Meeting Minutes

Participants:

CASAC Air Monitoring and Methods Subcommittee (See Roster with affiliations, Attachment A):

Mr. George A. Allen
 Dr. David T. Allen
 Dr. Linda J. Bonanno
 Dr. Doug Burns
 Dr. Judith C. Chow
 Dr. Kenneth Demerjian
 Mr. Eric Edgerton
 Mr. Henry (Dirk) Felton
 Dr. Philip Fine
 Dr. Philip Hopke
 Dr. Rudolf Husar
 Dr. Daniel Jacob
 Dr. Peter H. McMurry
 Dr. Allen Robinson
 Dr. Armistead (Ted) Russell
 Dr. James Jay Schauer
 Dr. Jay Turner
 Dr. Yousheng Zeng

Drs. David Allen, Linda Bonanno, Doug Burns, Phil Hopke, Daniel Jacob, Peter McMurry, James Schauer and Yousheng Zeng could not participate during the June 12, 2014 public teleconference.

EPA SAB Staff:

Mr. Edward Hanlon, Designated Federal Officer

Other Attendees:

A list of persons who requested information on accessing the public teleconference line is provided in Attachment B.

(a) EPA Meeting Minute

ATTACHMENT 2: ROSTER OF THE ACBCYW MEMBERSHIP

CHAIR

Elana Silber, MBA
 Executive Director
 Sharsheret
 1086 Teaneck Road
 Site 2G
 Teaneck, NJ 07666
 PHONE: 201-833-2341
 FAX: 201-833-25025
 E-mail: esilber@sharsheret.org
 Term: 1/17/2017 - 11/30/2020

MEMBERS

Lindsay Avner
 Founder and Chairman of the Board
 Bright Pink
 670 N. Clark Street
 Chicago, IL 60654
 Phone: 312-787-4412
 E-mail: LINDSAY@BEBRIGHTPINK.ORG
 Term: 1/18/2017 - 11/30/2020

Michele Maria Cerullo, JD
 Assistant General Counsel
 DaVita Medical Group, Legal Department
 10051 5th Street North, Suite 200
 St. Petersburg, FL 33702
 PHONE: 727-828-8910
 FAX: 727-258-0536
 E-mail: MMICERULLO@DVCLOUD.COM
 TERM: 4/01/2019 - 11/30/2020

EXECUTIVE SECRETARY

Temeika L. Fairley, PhD
 Designated Federal Official (DFO)
 Office of Program Development
 Division of Cancer Prevention and Control
 Centers for Disease Control and Prevention
 4770 Buford Highway NE., Mailstop F-76
 Atlanta, GA 30341
 Phone : 770-488-4518
 Fax : 770-488-4760
 E-mail : tf9@cdc.gov

Lisa Astalos Chism, DNP, APRN, NCMP, FAANP
 Clinical Director, Women's Wellness Clinic
 Nurse Practitioner
 Sexual Health Counselor and Educator
 Karmanos Cancer Institute
 4100 John R Street
 Detroit, MI, 48201
 Phone: 313-576-9326
 Fax: 313-576-8379
 E-mail: chismli@karmanos.org
 Term: 4/04/2019 - 11/30/2021

Anna Crollman
 Breast Cancer Advocate
 5021 Holly Brook Drive
 Apex, NC 27539
 Phone: 828-712-3706
 E-mail: mycancerchic@gmail.com
 Term: 4/08/2019 - 11/30/2020

(b) CDC Meeting Minute

and R to extract the name, affiliation, and position of each meeting participant from the FAC meeting minutes and generate structured CSV data. Table 2 shows the API prompt and R command we use. First, the prompt contains a phrase that asks GPT to create a delimited table of three columns. The prompt also contains sentences describing the information that GPT should fill in for each column based on the meeting minutes. Finally, the prompt asks GPT to clean the participants' names and remove commas that are not delimiters.

While GPT easily extracts individuals' names and affiliations, it often has difficulty extracting participants' position labels from meeting minutes because the labels are so diverse and broad. For example, "invited speaker" is not included as an example of a participant position in the prompt; as a result, GPT would often label those individuals as something else, such as "registered speaker." This could be problematic if accurately identifying individuals' positions is critical to understanding their roles in FAC meetings. To address this, researchers can include the extensive set of position labels that appear in meeting minutes in the prompt. However, we have also found that simply adding "etc." at the end of a list of example positions

quickly solves the problem by giving GPT the latitude to determine which information in the meeting minutes is about the participants’ positions.

Table 2: GPT Prompt and API Command in R

Description	Command in R
Prompt	<code>prompt= 'Return me a csv delimiter table of three columns, "name," affiliation," and "position." Do not return anything else except for the table. The first column "name" has the names of meeting participants and people, if any, who made public comments. When writing down names, remove any prefix, suffix such as Ph.D. or MPH, and texts within parentheses. The second column "affiliation" should have the information on people's affiliation. The third column should be labeled as "position" and specify whether people are "chair," "members," "Designated Federal Officer," "epa staff," "public participants," or "registered speakers," etc. Fill in all values for the "position" column. Remove all commas for values in columns. Use the following text to create the table:'</code>
Read PDF into R	<code>minutes= pdf_text("minutes.pdf") %>% str_split("_n")</code>
Run GPT API	<code>response= POST(url = "https://api.openai.com/v1/chat/completions", add_headers(Authorization = paste("Bearer", apiKey)), content_type_json(), encode = "json", body = list(model = "gpt-4-1106-preview", temperature = 1, messages = list(list(role = "user", content = paste(c(prompt,unlist(pdf_2022_4[eval(parse(text=print(meeting\$page[I]))])), collapse=" "))))))</code>
Extract API response	<code>capture.output(cat(content(response)\$choices[[1]]\$message\$content))</code>

While researchers may be concerned about data fabrication by GPT, we found that it rarely occurs in tasks like this, where GPT constructs datasets based on given information. After GPT created datasets from the meeting minutes, undergraduate RAs validated each dataset to ensure that all meeting attendee information was included in terms of names, affiliations, and positions. In our example, GPT failed to extract complete information from four of the 79 meeting minutes because our prompt did not include a complete list of participant positions. In this case, RAs helped fill in the position information for some participants that GPT was unable to retrieve from the meeting minutes.⁴

The collected data allow us to examine who attends these FAC meetings, which we show in Appendix B. The data show that a substantial number of interest groups voluntarily participate in FAC meetings and that their participation rate varies over time, which has not been documented by existing studies of federal advisory committees that focus primarily on FAC members appointed by agency heads (Feinstein and Hemel 2020).

4. Note that this success rate is based on running the same prompt again if GPT did not retrieve the complete information from the meeting minutes in the first trial. In R, this process can be automated by checking for missing information in the CSV columns.

Our example shows that the data collection and cleaning process for FAC meetings still requires human validation. However, having research assistants review the GPT-generated data is much less resource-intensive and time-consuming than hiring research assistants to build data based on meeting minutes. If the minutes of a meeting contains 50,000 characters (5-6 pages), it costs 30 cents to run the GPT code on the transcript.

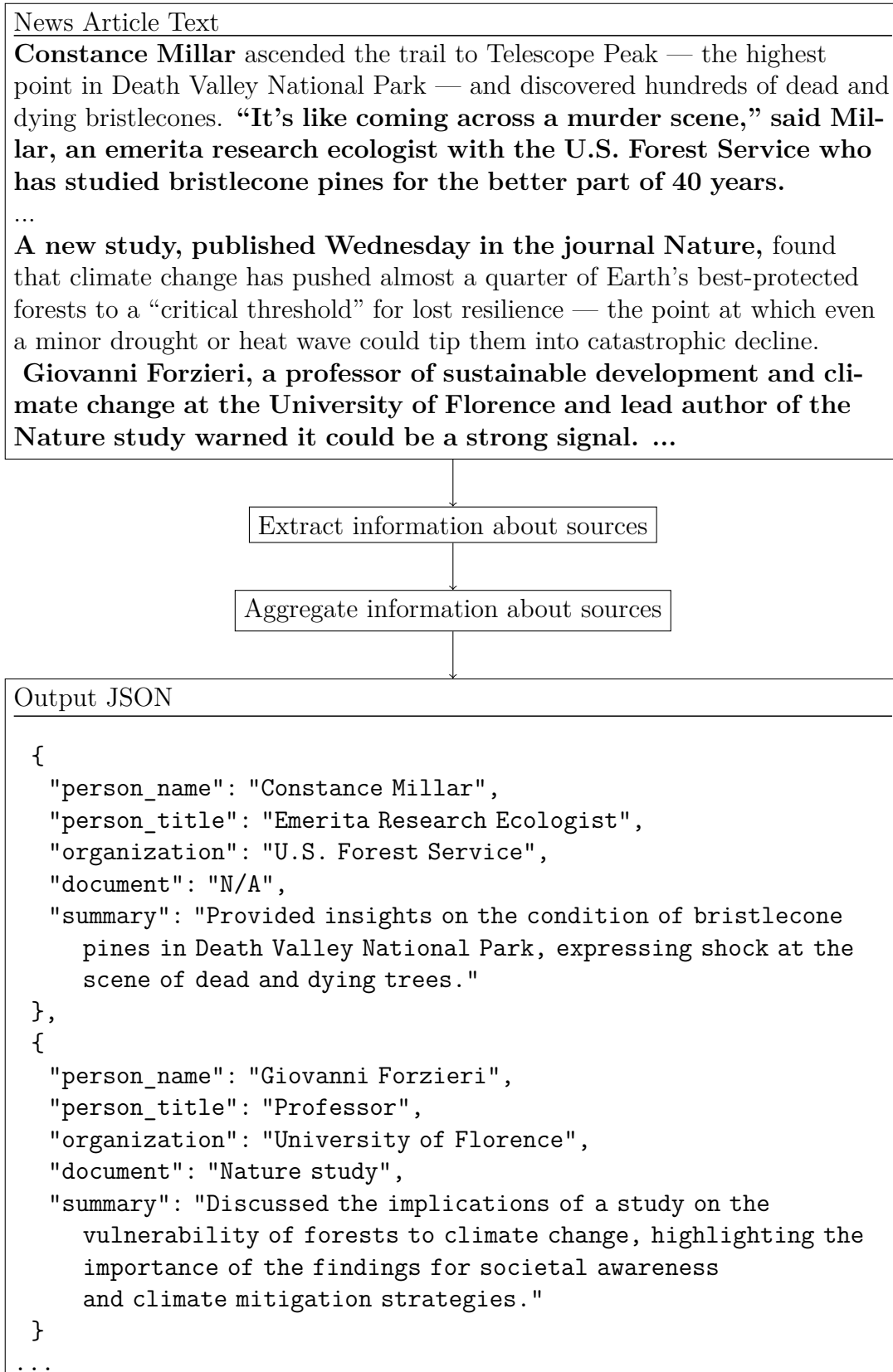
Example 3: Extracting Primary Sources from News Articles

In this section, we describe our approach to using GPT to extract semi-structured data from the extensive, unstructured text of news articles, with a focus on identifying the diverse sources cited by journalists. Newspaper articles typically reference a wide range of sources—from politicians and bureaucrats to private citizens and business owners—which significantly influences the information conveyed to the public. Although we focus on newspapers, our approach could be applied to similar tasks such as extracting witness information from court records or guest appearances in news transcripts.

Identifying sources is particularly challenging due to the length of the input documents and the nuanced integration of source information within the article text, including variations in name and context. In the initial phases of prompt development, we found that GPT had difficulty aggregating sources that were mentioned by multiple similar names and often failed to extract all sources, especially for longer articles. We suspect that this task is hampered by performance degradation as input text length increases the relatively complex level of reasoning required to identify and aggregate sources (Wei et al. 2023). Based on the common errors we observed, we divided the source extraction task into subtasks and used a separate prompt to solve each separately, with the output of one subtask prompt feeding directly into the next. This makes the logic of each subtask explicit, which also makes debugging easier.

The details of the method are shown in Figure 4. First, we identify all quotes and information attributed to third parties in the news article. Second, we aggregate the quotes

Figure 4: Source Extraction Process Outline

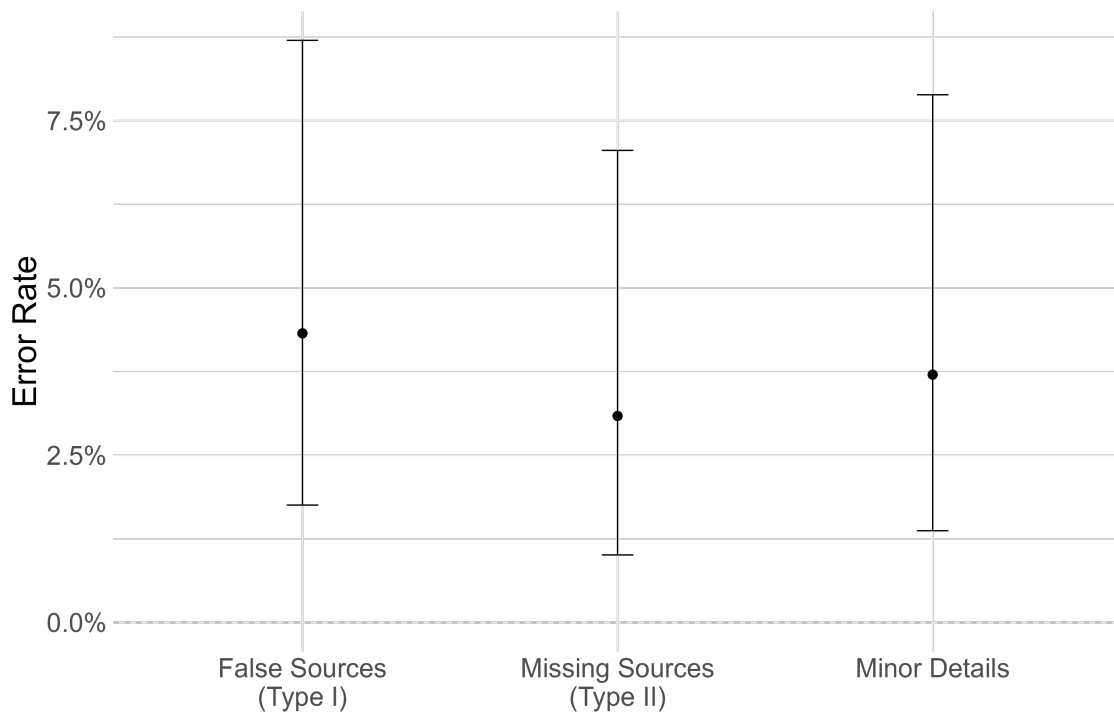


and information at the speaker or organization level. Finally, we transform the data into structured JSON that can be processed with any data tool of choice. The full set of prompts and sample output can be found in Appendix C.

To validate our approach, we used the described method to extract 214 sources for 50 articles and employed crowd workers to identify errors in the extracted sources. To ensure worker quality, we only include results from workers who successfully identified intentional errors that we embedded in the worker task (more details on crowd worker sourcing and screening are available in Appendix C). We identify three types of errors: minor details (i.e., incorrect title, name, or organization); false sources (Type I) where the extracted source was not cited in the article; and missing sources (Type II) where a source present in the article was not extracted. We manually review each error identified by the crowd workers and estimate the overall error rates. Our results show that the GPT-based system is highly accurate in extracting source details and rarely makes Type I or Type II errors (all error rates are less than 5%). Figure 5 shows the error rates with 95% confidence intervals. Furthermore, a manual inspection shows that the vast majority of the errors are edge cases, where it is difficult to determine with certainty the difference between a source citation and a mere mention of a particular entity (e.g., “President Xi Jinping of China has vowed repeatedly to move ahead with steps in his country to curb climate-altering pollution...”). In particular, when crowd workers noticed that source entities extracted by GPT were not cited in the article (Type I errors), these entities were always at least mentioned in the text. In other words, these errors were exclusively mistakes in judging whether a mentioned entity (i.e., Xi Jinping in the example above) should be considered a cited source, as opposed to outright hallucination of source entities. The remaining true missing source (Type II) errors tend to occur in longer articles with many (6+) sources.

We use this set of prompts to extract 31,431 sources from 5,795 *New York Times* articles about climate change over the period of 2012-2022 using the ‘GPT-4 Turbo’ model. Appendix Figure C1 shows the distribution of sources and articles per year. The total cost of the

Figure 5: Performance of GPT-based Source Extraction



extraction and validation was \$1,300.

Example 4: Extracting Elite Biographies from Online Sources

In this section, we leverage GPT to extract specific information from an unstructured corpus of sources obtained through systematic Google searches. This exercise reflects a broad category of data collection tasks where researchers cannot rely on a specific set of source material or a corpus of structured text. In these cases, data collection involves both searching for sources and extracting the relevant information. As a result, data collection draws from a variety of sources, such as websites, news articles, and academic and expert texts.

We replicate a large human-coded data collection effort by Montano, Paci, and Superti (2024), which examines whether having a daughter influences the pro-women policies of Italian mayors. The original study reflects a growing interest in political science in the role of elite biographical characteristics (Krcmaric, Nelson, and Roberts 2020). However,

this approach faces a significant challenge because systematic biographical data are rarely readily available. As a result, researchers must resort to time-consuming and expensive data collection. The original effort by Montano, Paci, and Superti (2024) leveraged systematic Google searches for 1,800 mayors. It was conducted by three research assistants from July 2023 to February 2024. For each mayor, the RAs reviewed up to the first 20 available search results for a total of over 7,300 Italian web pages.⁵ Each link was checked for three pieces of information: whether it contained any information about the mayor’s children, the number of kids, and the number of daughters.

We automated this process by scraping the original links and feeding the text into the GPT-4 Turbo API along with a carefully engineered prompt (see Table D1). We developed the prompt through an iterative trial-and-error procedure sampling random draws from the list of webpages and manually checking the model output. The final prompt includes instructions to make informational extraction more efficient, especially in edge cases. For instance, we directed GPT to infer gender from names and to assume the mayor has at least one child if they are mentioned to have grandchildren. Furthermore, since each webpage came from search results about a specific mayor, we could develop mayor-specific prompts, specifying their name and municipality.

This task tests GPT-4’s ability to parse through ambiguous and heterogeneous data. Most sources (about 90%) did not contain relevant information. The relevant information is encoded in myriad ways and the nuance of textual clues can be misleading. Table 3 shows illustrative examples of GPT-4 output. In three cases, GPT-4 correctly recovered the source information. The fourth case is an example of an error where the information is encoded in a complex way. The text mentions the mayor’s “only son” and his two daughters. GPT-4 understood this as the mayor having three children while in truth the two are the mayor’s son’s daughters and thus the mayor’s granddaughters, not to be counted as his direct

5. In subsection D.1 in the Appendix, we provide additional information on this process and how to automatize it.

offspring.

Table 3: Illustrative Examples of GPT-4 Information from Google Search Results

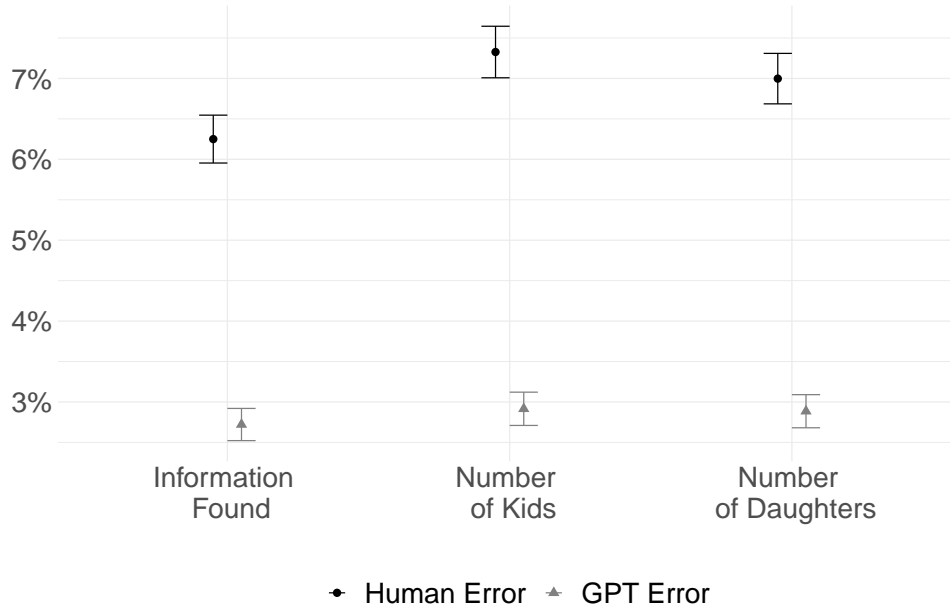
Source Text Relevant Mentions (Translated from Italian)	Extracted Data
<p>Success Case: Direct Information Encoding</p> <p>He lives in Gualdo Tadino with his partner Consuelo and their daughter Asia.</p>	<p>Information Found: 1 Number of Kids: 1 Number of Daughters: 1 Confidence: 90</p>
<p>Success Case: Indirect Information Encoding</p> <p>For me, these last few months have been full of surprises. The first, the most beautiful is the growth of my family which will soon expand.</p>	<p>Information Found: 1 Number of Kids: 1 Number of Daughters: NA Confidence: 80</p>
<p>Success Case: Complex Information Encoding</p> <p>As institutions and educational communities, we have a strong responsibility: to offer alternative and healthy models of sociality that allow our children to enjoy and rejoice in their age without exposing themselves to unnecessary risks.</p>	<p>Information Found: NA Number of Kids: NA Number of Daughters: NA Confidence: 100</p>
<p>Failure Case: Complex Information Encoding</p> <p>Alessandro Zanonato, 35 years old, is the mayor’s only son and lives with Chiara, a lawyer like him, and two daughters</p>	<p>Information Found: 1 Number of Kids: 3 Number of Daughters: 2 Confidence: 90</p>

Given the same set of search result links, we estimate the error rate of human coders and of GPT-4. We consider as ground truth all cases where human coders and GPT-4 agree. For all disagreements, we adjudicated between the two sets with a third round of human coding, with the help of new RAs. For cases where all three rounds disagreed—only seven out of the total sample—the authors personally hand-coded the ground truth.

Figure 6 illustrates the error rates against the ground truth by the original group of human coders and GPT-4. Across the three main pieces of relevant information, GPT-4 outperformed human coders. Figure D1 in the Appendix sorts the overall error rate into categories of mistakes: Type 1 (false positives), Type 2 (false negatives), and Type M (magnitude).⁶ Compared to human coders, GPT-4 makes fewer Type 1 errors and more Type 2 errors. On the one hand, this pattern is reassuring, since GPT-4’s output may not require extensive validation given its lower rate of false positives. On the other hand, it also suggests that GPT-4 may leave some information on the table, probably whenever it is encoded in an ambiguous or complex way.

6. Type M errors are adapted from Gelman and Carlin (2014) and refer to differences in magnitude between the ground truth and the collected information. For example, the number of children may be coded as 4 when the real number is 2.

Figure 6: Human Coders and GPT-4 Coding Error Rates



We also tested GPT-4’s ability to self-assess and found mixed results. The prompt asked GPT-4 to produce confidence ratings, on a scale of 0 to 100, about the accuracy of its output. The results are shown in Figure D2 in the Appendix. Whenever GPT-4 expressed a confidence rating below 100, the error rate increased significantly, from 2.8% to 27.3%. However, GPT-4 often expressed overconfidence, giving a rating of 100 to half of the errors found in this exercise. As such, confidence ratings can only be taken as a noisy indicator of potential error.

Limitations and Best Practices

Our four applications focus on data collection, cleaning, and extraction tasks that are tedious but common in quantitative political science research. These types of tasks allow for a straightforward application of LLMs while minimizing the potential for reasoning errors

and hallucinations. However, despite their straightforward nature, our applications also have limitations. In the context of data cleaning and collection, we also highlight important limitations and best practices. These recommendations integrate our experience and findings from our validation exercises in this study along with advice on emerging best practices for LLM use and prompt engineering (Ekin 2023).

First, LLM performance is highly sensitive to the specific prompt used. The term “prompt engineering” has emerged to describe the process of tailoring the LLM prompt to the task at hand. This task is iterative and potentially idiosyncratic to the specific application. However, general guidelines can improve the process. In our experience, the best performing prompts include several common components. To begin, the prompt should describe the task context, including the main objective and the type of input data. In addition, researchers should specify the output format, providing detailed descriptions of each data field. Prompts may also include examples of common information encoding patterns or even be computationally constructed to incorporate document-specific context. For complex tasks, we encourage researchers to explore multi-step prompts, as demonstrated in Example 3, or to ask the model to explain its reasoning before providing data as recommended by Wei et al. (2022).

Second, the context window of LLMs limits the length of both input and output text generated by the model. LLM performance also degrades as the text length increases even for documents that fit comfortably within the context window. Appendix Figure D3 shows that GPT makes more errors in identifying the mayor’s children as the length of the input text increases. A practical guideline is to limit texts to well under half of the advertised context window by selecting portions of the text that contain relevant keywords or by breaking tasks into smaller pieces.

Third, GPT occasionally does not follow the task instructions. This behavior can manifest as incomplete responses, incorrect column names, or incorrect data output. While prompt engineering can help mitigate these issues, we found that in most cases simply

re-running the same prompt multiple times until the output is well-formed is sufficient. Similarly, researchers can leverage logical dependencies across data fields to check for response coherence. For instance, in Example 4, we check that the number of children (of both genders) is greater than or equal to the number of daughters. A related concern is the production of “hallucinations” or false information. In our experience with data collection and cleaning tasks, outright hallucinations have not occurred. Researchers can experiment with the temperature parameter, which affects how much the LLM relies on current input data relative its training data. Lower values reduce the likelihood of hallucinations, but increase sensitivity to prompt wording and reduce reasoning ability. Temperature values range from zero to two, and we kept it below one in all of our examples.

Fourth, we make a few recommendations to improve the ergonomics of interacting with the GPT API. We recommend allowing the model to record the portions of the texts from which it extracts information. This addition can facilitate validation and shed light on the inner workings of the LLM information processing to aid in debugging. To simplify output data management, we recommend instructing the model to limit output to JSON or CSV/TSV format (“provide only the table and nothing else”).

Finally, we note the existence of competing LLMs in addition to OpenAI’s GPT. We focus on the GPT family of models in this paper because of their ease of use, widespread adoption, and high standard of performance. However, alternative models such as Google Gemini or Anthropic’s Claude, also may be worth considering.⁷ In particular, open-source models, such as Llama or Mistral offer significant cost and reproducibility advantages (Spirling 2023), but require more technical setup procedures and potentially lower generalized performance.

7. For more details on the differences between the models, see Korinek 2023.

Ethical Considerations

The use of LLMs raises ethical concerns related to professional, privacy, and environmental issues. Researchers should consider whether the potential costs of these novel tools outweigh the added efficiencies. Similarly, we encourage practitioners to consider strategies to limit or offset any negative downstream consequences of integrating LLMs into the research process.

First, the applications presented in this paper outsource tasks traditionally performed by student research assistants. While this improves the cost-effectiveness of data collection, it undermines student employment opportunities. These opportunities not only provide students with financial support but also provide valuable research experience and insight into academic work, potentially influencing some to pursue graduate studies. The research assistant experience strengthens students' résumés and also provides an important pedagogical opportunity for experiential learning. We encourage researchers to continue the practice of hiring promising students as research assistants. The use of LLMs does not completely eliminate the need for RAs, as validation still requires thorough human coding. Outsourcing repetitive data entry tasks to LLMs can free up time and resources to offer students more rewarding and intellectually stimulating tasks, such as exploratory literature reviews or more complex data management.

Second, LLMs raise potential privacy concerns. Given the rapid development of these models, no clear consensus has emerged on the confidentiality risks associated with input data (Wu, Duan, and Ni 2024; Yao et al. 2024). Therefore, we recommend that researchers exercise caution and avoid using the proposed techniques for any sensitive data.

Third, both research and anecdotal evidence show that LLMs may exhibit social biases embedded in their training data (Hida, Kaneko, and Okazaki 2024; Waight et al. 2024). As a result, information extraction tasks may produce output data that is consistent with the model's underlying biases, such as relying on stereotypes to decide ambiguous cases.

Researchers should evaluate whether their applications may be susceptible to this problem and focus validation efforts on detecting social biases in LLM output.

Finally, the development and operation of LLMs requires significant energy consumption, which raises environmental concerns Strubell, Ganesh, and McCallum 2020. Researchers might consider limiting their use of LLMs to cases where efficiency gains are clear and justify an increased environmental footprint. Similarly, for larger projects, researchers should evaluate the benefits of carbon offsetting strategies.

References

- Argyle, Lisa, Christopher Bail, Ethan Busby, and David Wingate. 2023a. “Leveraging AI For Democratic Discourse: Chat Interventions Can Improve Online Political Conversations At Scale.” *Proceedings of the National Academy of Sciences* 120 (41): e2311627120.
- Argyle, Lisa, Ethan Busby, Nancy Fulda, Joshua R Gubler, Christopher Rytting, and David Wingate. 2023b. “Out of One, Many: Using Language Models to Simulate Human Samples.” *Political Analysis* 31 (3): 337–351.
- Bisbee, James, Joshua Clinton, Cassy Dorff, Brenton Kenkel, and Jennifer Larson. Forthcoming. “Synthetic Replacements for Human Survey Data? The Perils of Large Language Models.” *Political Analysis*.
- Chiu, Ke-Li, Annie Collins, and Rohan Alexander. 2022. *Detecting Hate Speech with GPT-3*. arXiv: 2103.12407 [cs.CL]. <https://arxiv.org/abs/2103.12407>.
- Ekin, Sabit. 2023. *Prompt Engineering For ChatGPT: A Quick Guide To Techniques, Tips, And Best Practices*. <https://doi.org/10.36227/techrxiv.22683919.v2>.
- Feinstein, Brian D., and Daniel J. Hemel. 2020. “Outside Advisers and Inside Agencies.” *Georgetown Law Journal* 108 (5): 1139–1211.

- Gelman, Andrew, and John Carlin. 2014. “Beyond Power Calculations: Assessing Type S (Sign) and Type M (Magnitude) Errors.” *Perspectives on Psychological Science* 9 (6): 641–651.
- Hida, Rem, Masahiro Kaneko, and Naoaki Okazaki. 2024. *Social Bias Evaluation for Large Language Models Requires Prompt Variations*, arXiv:2407.03129, July. Accessed July 25, 2024. <https://doi.org/10.48550/arXiv.2407.03129>. arXiv: 2407.03129 [cs].
- Ji, Ziwei, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. “Survey of Hallucination in Natural Language Generation.” *ACM Computing Surveys* 55 (12): 1–38.
- Korinek, Anton. 2023. “Generative AI for Economic Research: Use Cases and Implications for Economists.” *Journal of Economic Literature* 61 (4): 1281–1317.
- Krcmaric, Daniel, Stephen C. Nelson, and Andrew Roberts. 2020. “Studying Leaders and Elites: The Personal Biography Approach.” *Annual Review of Political Science* 23 (May): 133–151.
- Montano, Beatrice, Simone Paci, and Chiara Superti. 2024. “The Mayor’s Daughter: How Patriarchal Norms Moderate The Daughter’s Effect on Mayoral Policy-Making.”
- Reed, Genna. 2021. “Administrator Regan Resets EPA Science Advice.” *The Equation*, accessed October 27, 2023. <https://blog.ucsusa.org/genna-reed/administrator-regan-resets-epa-science-advice/>.
- Spirling, Arthur. 2023. “Why Open-Source Generative AI Models Are An Ethical Way Forward For Science.” *Nature* 616 (April): 413.
- Strubell, Emma, Ananya Ganesh, and Andrew McCallum. 2020. “Energy and Policy Considerations for Modern Deep Learning Research.” In *Proceedings of the AAAI Conference on Artificial Intelligence*, 34:13693–13696. 09.

- Velez, Yamil Ricardo, and Patrick Liu. 2024. “Confronting Core Issues: A Critical Assessment of Attitude Polarization Using Tailored Experiments.” *American Political Science Review*, 1–18.
- Waight, Hannah C, Brandon M Stewart, Margaret E Roberts, Solomon Messing, Eddie Yang, Joshua A Tucker, and Yin Yuan. 2024. “Propaganda Bias: Why Training LLMs on Texts from State Media Matters.”
- Wang, Yu. 2023. “Topic Classification for Political Texts with Pretrained Language Models.” *Political Analysis* 31 (4): 662–668.
- Wei, Jason, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. “Chain-of-Thought Prompting Elicits Reasoning in Large Language Models.” *Advances in Neural Information Processing Systems* 35:24824–24837.
- Wei, Xiang, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, et al. 2023. *Zero-Shot Information Extraction via Chatting with ChatGPT*. <https://doi.org/10.48550/arXiv.2302.10205>. arXiv: 2302.10205 [cs].
- Wu, Xiaodong, Ran Duan, and Jianbing Ni. 2024. “Unveiling Security, Privacy, and Ethical Concerns of ChatGPT.” *Journal of Information and Intelligence* 2, no. 2 (March): 102–115.
- Yao, Yifan, Jinhao Duan, Kaidi Xu, Yuanfang Cai, Zhibo Sun, and Yue Zhang. 2024. “A Survey on Large Language Model (LLM) Security and Privacy: The Good, The Bad, and The Ugly.” *High-Confidence Computing* 4, no. 2 (June): 100211.
- Ziems, Caleb, William Held, Omar Shaikh, Jiaao Chen, Zhehao Zhang, and Diyi Yang. 2024. “Can Large Language Models Transform Computational Social Science?” *Computational Linguistics* (March): 1–55.

Online Supplementary Appendix

Table of Contents

- A Example 1: Cleaning and Analyzing Historical Data** **A2**
- A.1 Original Image and Text A2
- A.2 Comparison of OCR Results A3

- B Example 2: Extracting Unstructured Administrative Data** **A6**

- C Example 3: Extracting Primary Sources from News Articles** **A8**
- C.1 Full Prompts A8
- C.2 Validation Details A12
- C.3 Additional Figure A12

- D Example 4: Extracting Information from Unstructured Sources** **A13**
- D.1 Google Search Algorithm A13
- D.2 Full Prompts A13
- D.3 Additional Results A15

A Example 1: Cleaning and Analyzing Historical Data

A.1 Original Image and Text

Original Text (manually typed)

Two Negro soldiers and their wives complained at the Greyhound Bus Station in Atlanta that they had been mistreated by a bus driver while en route to Fort Benning, Ga. from Atlanta on 16 August. One of the Negro soldiers sat beside a white soldier with the latter's permission. When the bus driver observed the Negro's actions, he cursed him, made him move and, when the Negroes disembarked from the bus in protest at a stop near Atlanta, refused to return their tickets (C-3).¹

1. Weekly Intelligence Summary, 19 August 1944 to 25 August 1944, box 262, folder 2, Security Classified Reports and Memorandums Concerning Race Relations in the United States and Overseas, August 1944–January 1946, Record Group 107; National Archives Building, Washington, DC.

A.2 Comparison of OCR Results

In this section, we compare the performance of GPT with other OCR tools, including the freely available Tesseract, which is also the most prominent and widely used solution; the subscription-based Adobe Acrobat; and the cloud-based, paid OCR service, Google Cloud Vision (GCV). We first provide an example of the mixed quality of these tools using a single case from Record Group 107 (Appendix A.1), typed by the researchers. The results (Table A1) highlight the different levels of accuracy between these tools: text recognized by Tesseract and Adobe Acrobat is noisy, with multiple misspellings and unusual characters. While Google Vision typically provides near-perfect text recognition, using it to process thousands of images and waiting for results can be a time-consuming and sometimes unnecessary step, especially if the text has already been processed, albeit imperfectly.

Table A1: Comparison of OCR Results by OCR Tools

OCR Tools	Result*
Tesseract	<p>Two Negro. soldiers and their wives? complained. at the Greyhound ay ae . Bus Station in Atlanta that they had been mistreated by a bus driver i a F while en route to Fort Benning, Ge. from Atlanta on 16 4ugust. One ee: Be } of the Negro soldiers, sat beside a white soldier with the latter's [fo a cS permission, When the bus driver observed the . Negro'ts actions 16 as Ls. curséd him} made him move .and, whén the Negroes disembarked f -m the ' . fo Bae be bus in'protést at a stop near Atlanta, refused to return their i ie a tickets, _(C-3)</p>
Adobe Acrobat	<p>Two Negro. -oldiets: and their wiv-os.' complainc:. at the Greyhounrl Bus Ste.tion ;in b.tlanta that they had been mistreated ,by a bus driver while en rout.e: to Fort Benning, Ge . • from Atlanta on 16 /1.ugust. One of the l'1e::gro sol-diat; • ,pat beside a white soldier with the latter's permi;;ilon • . When the. bu · s dr:i.vGr bbserved the Negro's actions 1e curs, ;(.ma · e him - }nd, • ,1 • }1.n the Negro_es disembarked · m the bus 1n . protest at a stop · · · V.?near h.tla:nta, refused to return their ! .tickets •.C-3)</p>
Google Vision	<p>Two Negro soldiers and their wives complained at the Greyhound Bus Station in Atlanta that they had been mistreated by a bus driver while en route to Fort Benning, Ga. from Atlanta on 16 August. One of the Negro soldiers sat beside a white soldier with the latter's permission. When the bus driver observed the Negro's actions e curséd him, made him move and, when the Negroes disembarked from the bus in protest at a stop near Atlanta, refused to return their tickets. (C-3)</p>

* The errors are highlighted in bold text.

Table A2: GPT Prompt in R

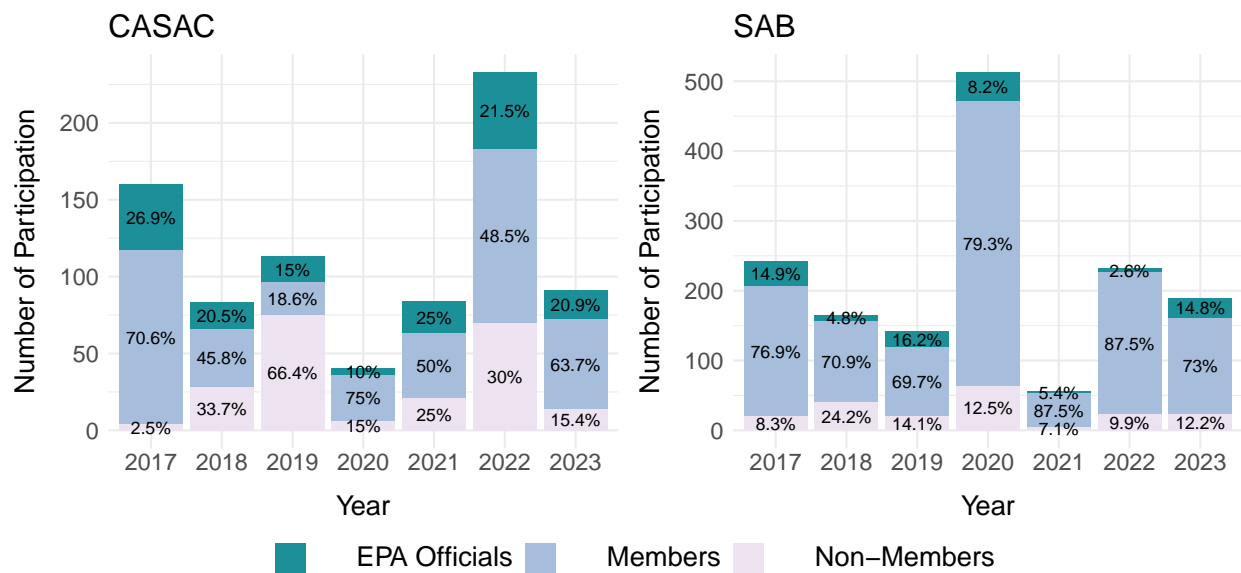
Description	R Code
Prompt	<pre>"This is a case of race-related incidents in WW2 US. Provide a csv table consisting of 3 columns. The column names should be: location, main actor, target. The columns are: (1) main actor: the person/organization who is doing the key action in the case (2) target: the person/organization to whom the action is targeted (3) location: the locality in which the case is happening:"</pre>

B Example 2: Extracting Unstructured Administrative Data

As an example, we focus on two FACs in the Environmental Protection Agency (EPA), which are the Clean Air Scientific Advisory Committee (CASAC) and Science Advisory Boards (SAB). CASAC provides independent advice to the EPA administrator on the technical basis for the EPA's National Ambient Air Quality Standards (NAAQS). To do so, the CASAC committee creates recommendation reports based on their reading of the EPA's drafts of Integrated Science Assessment (ISA) on NAAQS, EPA officials' presentations, and inputs from interest groups. SAB also reviews the quality and relevance of the scientific and technical information used by the EPA or proposed as the basis for agency regulations. The George W. Bush and Trump administrations attempted to fill many EPA advisory committees with their ideological allies, and the Biden administration reset EPA advisory committees in which the Trump administration intervened (Reed 2021).

Figure B1 shows the number and percentage of participation in FAC meetings. There are largely three types of participants: EPA agency officials who are career bureaucrats, members of FACs who are appointed by the EPA administrator, and interest groups that can voluntarily participate in FAC meetings and provide comments. If the same interest group participated in two FAC meetings in a given year, their participation counts as two in the statistics. The figure shows that the proportion of interest group participation in CASAC and SAB meetings is largest under the Trump administration when the administration was pursuing environmental policies against the preference of EPA officials. Thus, the data on FAC meetings provide an invaluable opportunity for researchers to examine how interest groups' participation in bureaucratic policymaking differs across different administrations.

Figure B1: Participation in CASAC and SAB Meetings, 2017-2023



C Example 3: Extracting Primary Sources from News Articles

C.1 Full Prompts

Table C1: Extracting Sources from News Articles with GPT: Identifying Quotes and Information (Step 1)

1: Quotes and Information

System: You are a research assistant whose task is to extract quotes and other external information used by journalists in news articles.

The user will provide the text of a news article.

Generate a numbered list of all quotes and external information attributed to specific people, organizations, or documents such as studies, reports, or press releases. Format your response as follows:

1. Quote or information - Name of the source - Background of the source - Context of the quote or information

Some articles do not mention any information drawn from external sources. In these cases, simply say “No sources mentioned.”

User: [Original News Article Text] The trees had stood for more than 1,000 years. The rings of their trunks told the story of everything they’d witnessed. Weather patterns shifted; empires rose and fell. But here, in one of the harshest environments on the planet, the bristlecone pines survived.

Until the day in 2018 when Constance Millar ascended the trail to Telescope Peak — the highest point in Death Valley National Park — and discovered hundreds of dead and dying bristlecones.

“It’s like coming across a murder scene,” said Millar, an emerita research ecologist with the U.S. Forest Service who has studied bristlecone pines for the better part of 40 years.

In a study published this spring, she and fellow researchers showed that the West’s worst drought in at least 1,200 years had critically weakened the trees. Voracious bark beetles — a threat to which bristlecones were previously thought immune — delivered the death blow.

Standing amid the devastation atop Telescope Peak, Millar realized, “This could be a harbinger of what could be happening in the future.” Millar immediately contacted Barbara Bentz, a research entomologist for the Forest Service based at the Rocky Mountain Research Station in Logan, Utah.

Just a few years earlier, Bentz had surveyed bristlecone populations across the West and found that few were succumbing to the bark beetle outbreaks that decimated so many other species. She felt confident that the trees would remain resilient even as rising temperatures caused beetle populations to explode.

Yet, when Bentz peeled a piece of bark off one of the Death Valley bristlecones, she found scores of tiny tunnels created by beetle larvae as the insects chewed through the tree’s living tissue.

“Oh, no,” Bentz thought to herself. “It really is beetles. Oh, no.”

She and her colleagues uncovered further evidence of insect damage in Utah’s Wah Wah Mountains, where young bristlecones were attacked by a small brown beetle called a pinyon ips.

A new study, published Wednesday in the journal Nature, found that climate change has pushed almost a quarter of Earth’s best-protected forests to a “critical threshold” for lost resilience — the point at which even a minor drought or heat wave could tip them into catastrophic decline.

“It’s a strong warning, I think, for society,” said Giovanni Forzieri, a professor of sustainable development and climate change at the University of Florence and lead author of the Nature study.

He pointed out that most of the world’s climate plans count on forests to pull planet-warming gases out of the atmosphere. If these ecosystems collapse, humanity will find it difficult, if not impossible, to stave off catastrophic warming.

Response: 1. “It’s like coming across a murder scene.” - Constance Millar - Emerita research ecologist with the U.S. Forest Service, with over 40 years of experience studying bristlecone pines. - Context: Describing her reaction upon discovering the dead and dying bristlecone pines in Death Valley National Park.

2. “This could be a harbinger of what could be happening in the future.” - Constance Millar - Emerita research ecologist with the U.S. Forest Service. - Context: Reflecting on the implications of the dying bristlecone pines for future forest health.

3. “Oh, no. It really is beetles. Oh, no.” - Barbara Bentz - Research entomologist for the Forest Service based at the Rocky Mountain Research Station in Logan, Utah. - Context: Expressing her dismay upon finding evidence of beetle larvae damaging bristlecone pines in Death Valley.

4. “It’s a strong warning, I think, for society.” - Giovanni Forzieri - Professor of sustainable development and climate change at the University of Florence and lead author of the Nature study. - Context: Commenting on the findings of a study published in Nature regarding the vulnerability of the world’s forests to climate change and the implications for climate plans that rely on forests to sequester carbon.

Table C2: Extracting Sources from News Articles with GPT: Source Aggregation (Step 2)

2: Source Aggregation

System: You are a research assistant whose task is to understand the sources of quotes and other external information used by journalists in writing news.

The user will provide a list of quotes and external information referenced in a news article.

Aggregate together the quotes and information drawn from the same sources into a new numbered list. Provide a description of the source and a summary of the source's contribution. Format your response as follows:

1. Name of the source - Description of the source - Summary of the source's contribution - Documents authored by the source, if relevant

User: [List of quotes and information from output of Step 1]

Response: 1. Constance Millar

- Description of the source: Emerita research ecologist with the U.S. Forest Service, with over 40 years of experience studying bristlecone pines.

- Summary of the source's contribution: Constance Millar provided insights on the condition of bristlecone pines in Death Valley National Park, expressing shock at the scene of dead and dying trees and suggesting that this situation may foreshadow future forest health challenges.

2. Barbara Bentz

- Description of the source: Research entomologist for the Forest Service based at the Rocky Mountain Research Station in Logan, Utah.

- Summary of the source's contribution: Barbara Bentz shared her reaction to discovering beetle larvae in bristlecone pines, indicating concern for the trees' health and the potential impact of pest infestations.

3. Giovanni Forzieri

- Description of the source: Professor of sustainable development and climate change at the University of Florence and lead author of the Nature study.

- Summary of the source's contribution: Giovanni Forzieri discussed the implications of a study on the vulnerability of forests to climate change, highlighting the importance of the findings for societal awareness and climate mitigation strategies.

- Documents authored by the source: Lead author of the study published in Nature on the vulnerability of the world's forests to climate change.

Table C3: Extracting Sources from News Articles with GPT: Organization and JSON (Step 3)

3: Structured Data

System:

For each entry in the given list of sources, identify the person's name (if the source is a person), their title (such as President, Professor, Spokesman, etc.), the organization name, the document name (for sources that are written documents such as academic studies, press releases, books, etc.), and the category (already provided). If any of these are not relevant provide a value of N/A. For unnamed or anonymous sources provide a value of "Anonymous" for the person's name. Do not include titles in the person name field (eg, Barack Obama instead of President Obama). Finally, include the summary of the primary source's role in the article.

Format your response as JSON array with keys "person_name", "person_title", "organization", "document", "category" and "summary"

User: [List of aggregated sources from output of Step 2]

Response:

```
[
  {
    "person_name": "Constance Millar",
    "person_title": "Emerita Research Ecologist",
    "organization": "U.S. Forest Service",
    "document": "N/A",
    "summary": "Provided insights on the condition of bristlecone pines in
    Death Valley National Park, expressing shock at the scene of dead and dying trees."
  },
  {
    "person_name": "Barbara Bentz",
    "person_title": "Research Entomologist",
    "organization": "Forest Service Rocky Mountain Research Station",
    "document": "N/A",
    "summary": "Shared her reaction to discovering beetle larvae in bristlecone pines,
    indicating concern for the trees' health."
  },
  {
    "person_name": "Giovanni Forzieri",
    "person_title": "Professor",
    "organization": "University of Florence",
    "document": "Nature study",
    "summary": "Discussed the implications of a study on the vulnerability of forests
    to climate change, highlighting the importance of the findings for societal awareness
    and climate mitigation strategies."
  }
]
```

C.2 Validation Details

We collected corrections for 462 sources for 87 articles from 25 crowd workers hired on the Prolific platform with an undergraduate or graduate degree and English fluency. In the main text, we report data based on results that remove data from inattentive crowd workers, resulting in a smaller validation set of 241 sources. In order to determine crowd worker attentiveness, we asked each crowd worker to correct errors for a “test” article that contained manufactured errors known to us in advance. This test article was embedded in the task, making it impossible for the crowd worker to determine whether there was a test article or which article it was. “Inattentive” workers were those who did not catch these manufactured errors. Thus, we analyze corrections made by eight attentive crowd workers, who each read two articles and made corrections on a total of 214 sources extracted by the GPT-based system. Error rates are even lower if we include data from inattentive workers (ranging from 1.1% and 1.5% for the three error types).

C.3 Additional Figure

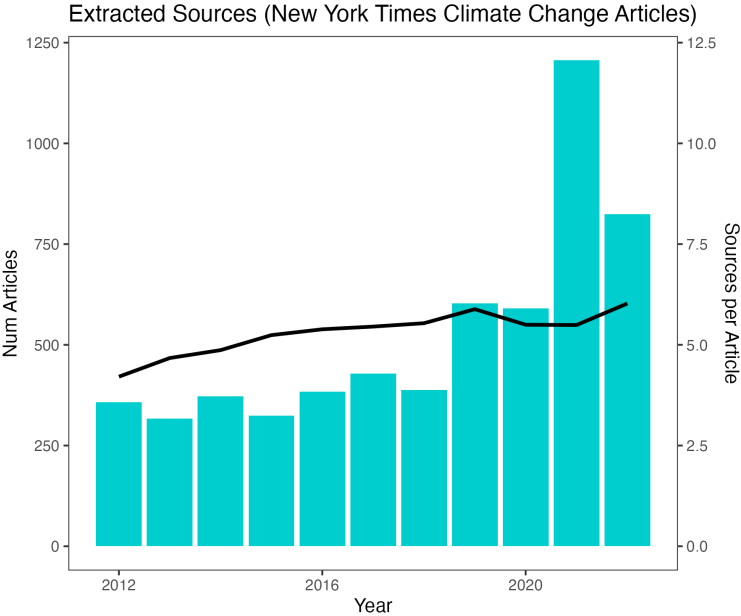


Figure C1: We extract 31,431 sources from 5,795 *New York Times* articles about climate change over the course of 2012-2022.

D Example 4: Extracting Information from Unstructured Sources

D.1 Google Search Algorithm

The first part of the data collection in Montano, Paci, and Superti (2024) relies on systematic Google searches. This procedure offers several key advantages. Relying on search engine results is highly flexible and can be applied to various data collection projects that rely on publicly available information. It can collect a wide range of sources, thus triangulating information retrieval for topics discussed across areas of the internet. Alternatively, search strategy can be refined to rely on specific search engines, such as Google News or Google Scholar, or to target only specific types of sources, such as institutional websites or blogs.

At the same time, this strategy has clear drawbacks. Source selection depends on the internal algorithm of the chosen search engine and opaque dynamics such as SEO (Search Engine Optimization) indexing. This may skew search results toward sources with greater resources and web visibility. In addition, the underlying information base is constantly changing as websites are shut down and search results change. As a result, the data collection process is difficult to replicate. However, researchers can ensure source traceability by maintaining a database of the underlying sources, such as text scraped from search result links.

This first step can also be automatized. Relying on the JSON Google Search API, we created an algorithm that iterates over predetermined combinations of search terms, collecting the top ten search result links for each search. We then scrape the resulting set of web pages for their text content. The full R code is available upon request.

D.2 Full Prompts

Table D1: Full Prompts

Example 4 - Elite Biographical Information Extraction

In this task, you will read an Italian text scraped from Google results; it could be any webpage. Search the text for specific information, and return to me a csv delimiter table, using commas as separators, with the following columns:

“info-found,” “has-kids,” “number-kids,” “number-daughters,” “source,” “confidence.”

Do not return anything else except for the table.

In the text I provide below, search for information about whether [NAME], mayor of [TOWN], has kids, how many kids the mayor has, and how many of the kids are female.

Be careful, the text may discuss the kids of other people and also mention the mayor. It is not enough to have the word “figli” and the mayor name. The text may also discuss other mayors of [TOWN]. Make sure that the text attributes the kids mentioned to mayor [NAME]. The text may also mention [NAME] as former mayor of [TOWN], in this case you can still consider the information as valid. The text may also connect the kids to the mayor indirectly, discussing the broader family of the mayor, but make sure that there is a clear link. Also, if the mayor is said to have a granddaughter or grandson, assume they must have at least one kid. In this case, unless it is mentioned which parent is the mayor’s child, assume the child is male.

The column “info-found” should have value 1 if you find any information on any of these queries. If you find no information about kids of the mayor, the column should have value 0.

The column “has-kids” should have value 1 if you found information about the mayor having kids. Input 0 if there is information about the mayor not having kids. Otherwise, input NA.

The column “number-kids” should have a numeric value for the number of kids, both male and female. If there is a specific number of kids, use that number. If the text specifies kid’s names, use the number of names. Otherwise, input NA.

The column “number-daughters” should have a numeric value for the number of daughters. In case you can find the names of the daughters, infer the gender from the names. In Italian, most names have a clear gender connotation. If there is a specific number of daughters, use that number. If the text specifies daughter names, use the number of names. Otherwise, input NA.

The column “source” should have the sentences where you found information for any of the values imputed in the previous columns, pasted together with a semi-colon separating them. If you did not find info, input NA.

The column “confidence” should contain a number from 0 to 100 indicating the degree of confidence in the information extraction for all previous columns. Give a rating both when you find and when you do not find information.

Follow these guidelines to determine confidence:

Clarity of Attribution: Increase confidence if the information about kids clearly refers to [NAME]. Decrease if other individuals could be the parents.

Contextual Evidence: Increase confidence if multiple sources within the text confirm the information. Decrease if the information comes from ambiguous or unreliable parts of the text.

Similar Names: Decrease confidence if there are multiple people with the same name or if the text discusses other mayors.

Conflicting Information: If the text contains conflicting information regarding the kid, decrease confidence significantly.

Use the whole scale, following these descriptions of different confidence ranges:

0-20: High ambiguity or multiple conflicting sources.

21-40: Moderate ambiguity, or some elements are unclear but likely correct.

41-60: Slightly unclear, minor contradictions or uncertainties present.

61-80: Mostly clear, with minor unresolved questions.

81-100: Clear and unambiguous information, strongly supported by the text.

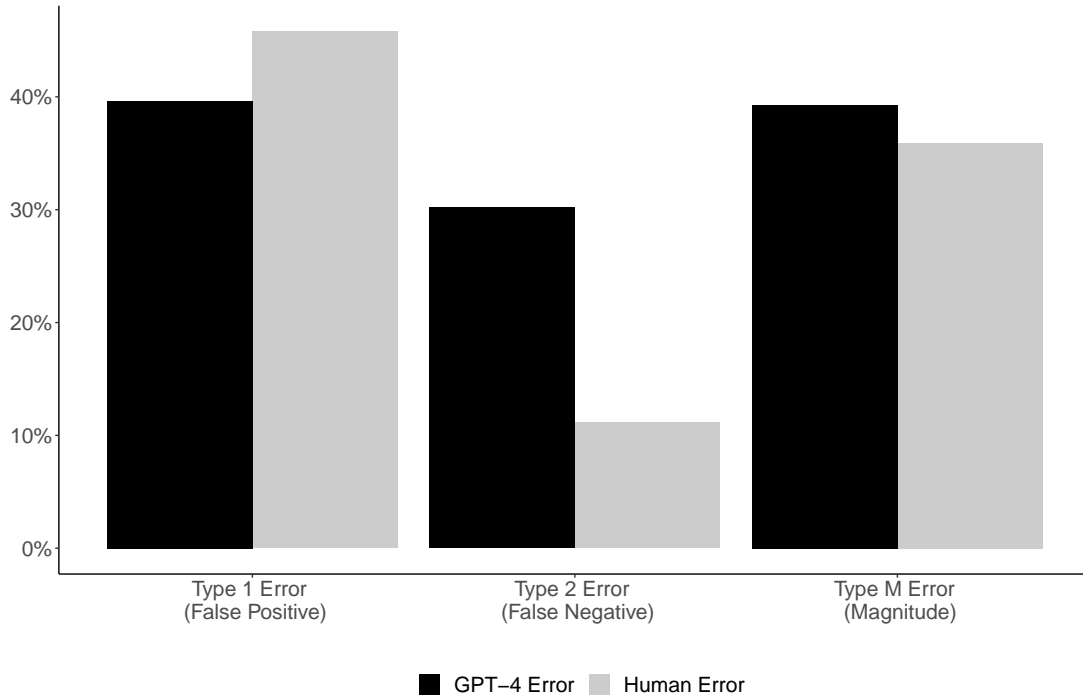
These ranges are just for reference, the confidence rating should have a number from 0 to 100, not a range.

This is a really important task, and you will be rewarded if you do a great job. So put maximum effort into it, please!

Use the following text to create the table: [Insert text scraped from web page]

D.3 Additional Results

Figure D1: Error Types



Error rates are calculated as the number of errors of a specific type over the number of sources checked.

Figure D2: GPT Confidence Rating and Coding Error

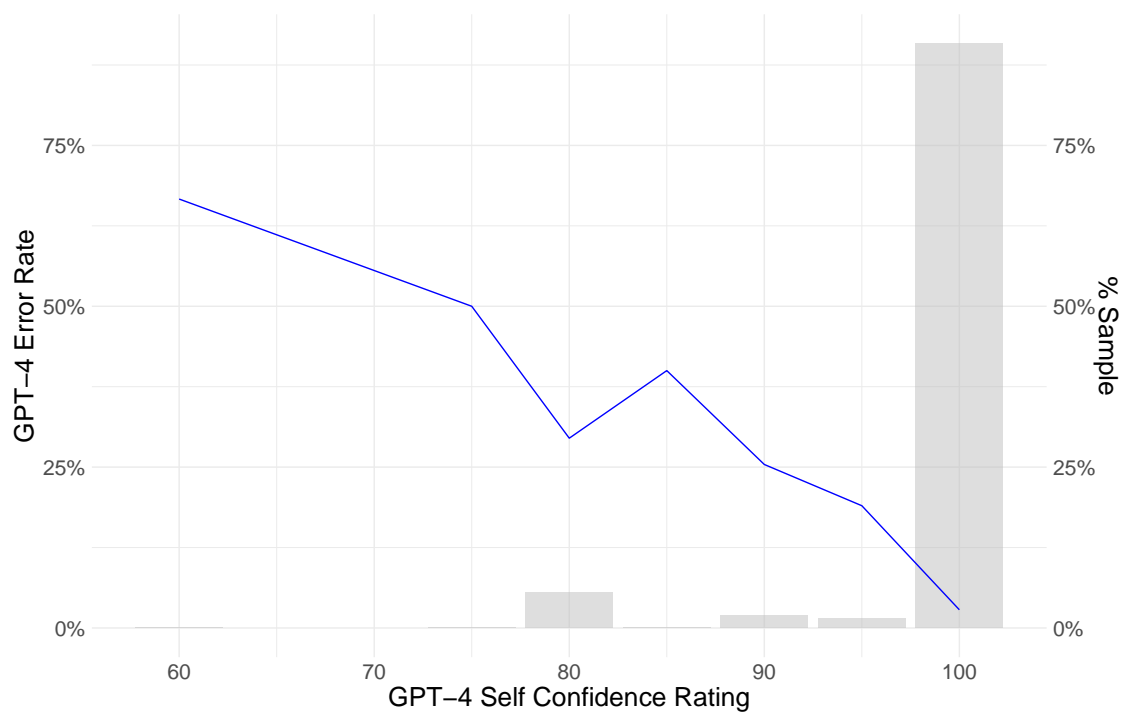


Figure D3: Source Text Length and Coding Error

