# From Codebooks to Promptbooks: Extracting Information from Text with Generative Large Language Models

*Oscar Stuhler, Northwestern University*
*Cat Dang Ton, Northwestern University*
*Étienne Ollion, Centre national de la recherche scientifique*

**Abstract**

Generative AI (GenAI) is quickly becoming a valuable tool for sociological research. Already, sociologists employ GenAI for tasks like classifying text and simulating human agents. We point to another major use case: the extraction of structured information from unstructured text. Information Extraction (IE) is an established branch of Natural Language Processing, but leveraging the affordances of this paradigm has thus far required familiarity with specialized models. GenAI changes this by allowing researchers to define their own IE tasks and execute them via targeted prompts. This article explores the potential of open-source large language models for IE by extracting and encoding biographical information (e.g., age, occupation, origin) from a corpus of newspaper obituaries. As we proceed, we discuss how sociologists can develop and evaluate prompt architectures for such tasks, turning codebooks into "promptbooks." We also evaluate models of different sizes and prompting techniques. Our analysis showcases the potential of GenAI as a flexible and accessible tool for IE, while also underscoring risks like non-random error patterns that can bias downstream analyses.

**Acknowledgments:**

The authors are grateful to Émilien Schultz for helping them with advanced data manipulation. We also thank the participants of the NLP & Social Sciences seminar at the Institut Polytechnique de Paris and those of the Social Science Colloquium at the University of Stuttgart for their insightful comments.

**Corresponding author:**

Oscar Stuhler
Northwestern Department of Sociology
1812 Chicago Ave
Evanston, IL 60201
Email: oms@northwestern.edu

# Introduction

Generative artificial intelligence (GenAI) is increasingly recognized as a powerful tool for humanities and social sciences research (Bail 2024; Davidson 2024). In particular, generative large language models (LLMs)[1] have been used to simulate human behaviors across a variety of roles and contexts (Gao 2023; Park et al. 2023; Törnberg 2023; Broska, Howes, and van Loon 2024; Kozlowski and Evans 2024), such as economic agents in markets (Horton 2023), subjects in psychological experiments (Aher, Arriaga, and Kalai 2023), opinion poll respondents (Argyle et al. 2023; Kozlowski, Kwon, and Evans 2024), college applicants (Alvero et al. 2024), or talk show participants (Karell, Sachs and Barrett 2023). More prosaically, but no less importantly, GenAI has been used to aid question generation (Götz et al. 2023), missing data imputation (Kim and Lee 2024), and content classification (Gilardi et al. 2023; Chae and Davidson 2024; Davidson 2024; Law and Roberto 2024).

Adding to this growing body of work, this paper explores the potential of GenAI, specifically generative LLMs, as a tool for information extraction (IE). IE is an umbrella term for a variety of Natural Language Processing (NLP) methods for finding information in unstructured textual data (Grishman 2022; Jurafsky & Martin 2024 Ch. 20). For instance, given a collection of short biographies, an IE model may be trained to extract the subject's date of birth, birthplace, and name to fill a template with this information. IE is a relatively wide field that includes such diverse tasks as temporal tagging, event extraction, named entity recognition, or relation extraction. While sociologists have long sought to systematically extract specific kinds of information from texts (see, e.g., Franzosi 1989; Mohr 1994; Tilly 1995), few to date have made use of IE-family approaches (recent exceptions include Knight 2022; Goldenstein and Poschman 2019; Mohr et al. 2013; Stuhler 2021, 2022, 2024). This is largely because, thus far, doing IE has required familiarity with a branch of highly specialized NLP methods, but also because such models may not have aligned with how sociologists want to code their data.

In this article, we make the case that the advent of GenAI fundamentally changes this. Specifically, we illustrate the potential of prompt-based IE in a study of newspaper obituaries, from which we encode a variety of biographical information. Alongside this empirical analysis, we discuss how social scientists can develop and evaluate prompt architectures for IE, turning codebooks into what we call "promptbooks." We find that the most capable model we tested (Llama 70B Instruct) replicates our manual extractions with a high degree of accuracy for most types of information. Especially information that is explicitly stated in the text (e.g., *age* or *cause of death*) and that requires numeric inference (e.g., the *number of children*) was extracted with near-perfect accuracy. However, accuracy declined somewhat for information requiring

[1] We emphasize that LLM stands for large language model only and that there are LLMs that are not generative.

interpretive competence (*education level*, *origin*, *religion*). In these cases, the model at times adopted an overly literal reading, struggling to integrate contextual cues into broader inferences. When testing different model sizes, we found that a smaller model (Llama 8B Instruct) could perform as well as the larger one for some variables, but not others. Different prompting strategies (chain-of-thought, 0- versus 1-shot, requesting JSON-formatted responses) made little impact on performance.

Overall, our analyses showcase some core advantages of prompt-based IE for social scientists but also one major risk. Firstly, the approach is highly *accessible*. In fact, we believe researchers with basic coding skills will be able to build reasonably accurate IE systems. Second, it is highly *flexible*: thus far, social scientists who wanted to do IE have been limited by whether or not computational linguists had built modules that fit their research problem. With prompt-based IE, social scientists can design their own tasks and extract information relevant to their research context and questions. Third, the approach has the potential to enhance *transparency* and *reproducibility* in computational research. In theory, codebooks for content analysis should be sufficient to reproduce the classification they are designed for, but in practice, they often depend on at least some implicitly shared knowledge among the coders. Prompt-based IE forces researchers to explicitly and more rigorously define the qualities they are aiming to measure. Notwithstanding these advantages, our results also suggest that prompt-based IE comes with a major risk for social science: the *non-randomness of errors*. Models tend to provide plausible-sounding responses and, in doing so, sometimes infer them based on semantic patterns learned from their training data, leading to potentially biased predictions. Even when extracted information is highly accurate in the aggregate, this can create serious pitfalls for downstream analyses.

The paper is structured as follows: first, we define IE and discuss its relevance and prior applications in sociological research. Second, we introduce our data and argue why obituaries offer a good case study for IE. Third, we define a diverse set of extraction tasks and discuss how we developed a *promptbook* for them. Our goal here is not so much to present the reader with the eventual product, but rather to shed light on our research process. Fourth, we evaluate our approach for different variables and test out the relative performance of different prompting strategies and model sizes. We then summarize our observations and offer guiding thoughts for researchers. Finally, we conclude with a discussion of the opportunities and risks of using prompt-based IE for sociological research.

## What is information extraction, and how does it matter for sociology?

Information extraction refers to the process by which the unstructured information embedded in texts is identified and turned into structured data (for introductions, see Grishman 2022; Jurafsky & Martin 2024, Ch. 20). Texts typically contain rich information about events, people, organizations, places, relations, and other kinds of entities. Yet this information remains unusable for systematic analysis unless it is harvested and stored in a database. An IE engine makes this information tractable by parsing unstructured text to identify and extract claims about specific events or relations, following pre-defined templates.

We offer an illustration of this in Figure 1. Imagine, for instance, one would like to study market dynamics based on events reported in business news, but these news reports typically contain a mix of important and relatively redundant information. An IE engine specializing in parsing business news might have a template for acquisition events that includes the elements [acquiring company], [target company], and [deal value]. The engine's task is then to identify such events and fill this template with information from the reports. Similarly, the same engine might also have a template for identifying leadership positions with the elements [person], [role], and [company].

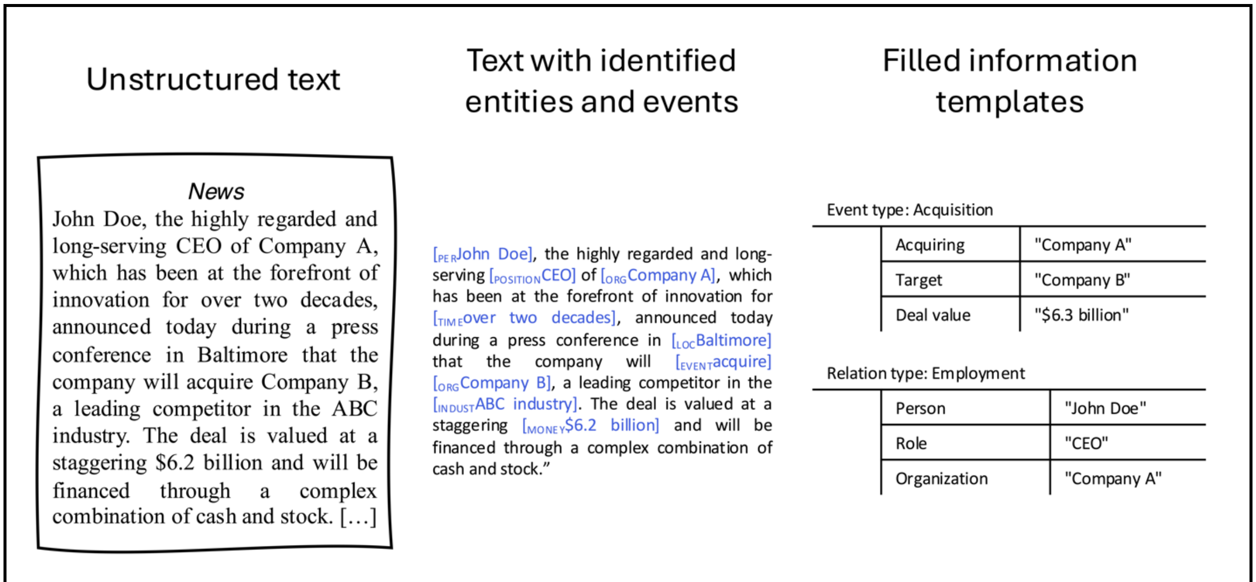


**Figure 1. Illustration of Information Extraction Template Filling.**

Over the past three decades, NLP scholars have built IE systems for a variety of applications (for a review, see Grishman 2019). Sub-tasks of an IE system often include named entity recognition, temporal tagging, event extraction, and relation extraction. IE systems are typically built for specific domains and specialize in extracting the events or semantic relations important to that domain. For instance, a system built for business news (e.g., Jacobs & Hoste 2022) might be trained to detect acquisitions, merger events, collaborations (e.g., of the form collaborate{Company A, Company B}), or leadership transitions (e.g., replace{Person A, Person B, Company, Position}). Meanwhile, a system specializing in medical language (e.g., Wang et al. 2018; Koleck et al. 2019) would detect claims about the effects of specific substances (e.g., cause{Pharmacological substance, Pathological function}).

We note three important conceptual clarifications here. First, the term "information" in NLP scholarship is understood relative to the text and carries no claims as to the truth value of what is extracted. Put more precisely, IE's aim is to create formalized representations of information *asserted* in a text. Second, while the exact line between these concepts is not always easy to define, IE is conceptually distinct from *text classification*, where the goal usually is to assign one or more labels (categories) to an entire text based on its overall content or theme. Meanwhile, IE typically involves detecting granular, localized pieces of information scattered throughout the text. Third, IE rests on having clearly defined conceptual templates. These should define among other things the possible values (also referred to as "slot fillers") for each field and the criteria for assigning them. Values can be either text segments (e.g., a company name) or specific concepts that can be inferred from the text and that were pre-specified in the template (e.g., a specific organization can be classified as *governmental*).[2] For a more detailed discussion, we refer to Jurafsky and Martin (2024, see esp. p. 22-24).

IE has evident potential for sociological research. While not using NLP methods, sociologists have long engaged in a similar form of template filling in which textual data are treated as informants on real-world events and relations. The earliest prominent example of this is research that creates event catalogs from newspaper data. Franzosi (1989) was among the first to propose a subject-action-object coding scheme, which he used to extract information about industrial conflict and protest events from Italian newspapers (also see Franzosi 1990 and 1994). This approach was later taken up and expanded by Tilly (1997), who extracted information on more than 8,000 contentious gathering events from a corpus of 18th-century newspapers.

Perhaps the most impressive work in this style stems from the Dynamics of Collective Action Project (see e.g., Wang and Soule 2012, 2016), which created a database of 23,000 protest events in the United States (1965-1995). To do this, researchers examined the *New York Times* for mentions of protest events. When a protest event was mentioned, researchers recorded a range of information, including protest size, location, primary target, tactics used by protestors, duration, police presence, potential violence, and more, if available. Following a similar approach, Braun and Koopmans (2014) used an event coding template that includes bystander reactions to study their effects on xenophobic attacks in Germany (also see Braun and Koopmans 2010). Makovi et

[2] For an impressive example of what defining such templates can look like, we refer to the collected ACE Tasks and Specifications codebooks by the Linguistic Data Consortium (2024).

al. (2016) code newspaper accounts of lynchings using a template that includes information on place, time, race and sex of the victims, the justification given for the mob's action, the racial makeup of the mob, whether or not an intervention occurred, as well as whether it succeeded, and what kind of actor undertook it.

Beyond cataloging events, sociologists have used similar information templates for recording a variety of information from text, including political claims (Mische & Pattison 2000), practices or attributes associated with specific identities (e.g., Mohr 1994; Mohr and Duquenne 1997; Mohr and Lee 2000; Martin 2000; Carley 1993, 1994), or causal links between events (Bearman and Stovel 2000; Bearman et al. 1999; Smith 2007). While these different analyses extract information and fill templates much like NLP-based IE models do, the extraction was primarily done by hand and often required an immense investment of labor and research money (for a more systematic review of this literature, see Stuhler 2022).

Only more recently, sociologists have begun to embrace IE-adjacent computational methods in their research, including a growing excitement about syntax-based parsers. Mohr and colleagues (2013) combined dependency parsers and named entity recognition in their analysis of the National Security Strategy documents. Knight (2022) used dependency parsers to analyze the kinds of actions attributed to corporate entities in early 20th-century newspaper articles. Similarly, Goldenstein and Poschmann extracted subject-verb-object triplets from business news to study corporate responsibility discourse (2019). Also relying on dependency parsers, Stuhler has analyzed news coverage of immigration in Germany (2021), as well as the relationship between gender and agency in fiction writing (2024).

Parsers grounded in syntax, however, differ slightly from the kind of IE approach we focus on in this paper, in that they use a general grammar (e.g., say, actor, action, recipient) instead of the more specific information templates exemplified in Figure 1. This has the advantage of allowing the extracted content to stay close to the original text without requiring researchers to make rigid a priori assumptions. Inductive methods can then be used in a second step to uncover structures in this content. This is important for sociologists who, for good reason, hesitate to adopt pre-defined coding schemes and put a premium on having patterns emerge inductively from the data. For instance, Stuhler's (2021) analysis of refugee discourse in newspapers involves extracting a series of basic elements (actions, attributes, and possessions) around mentions of refugees, then inductively identifying latent conceptions of refugees using a clustering algorithm. It is worth noting that one possible reason why sociologists have not embraced existing IE systems is that such systems typically assert rigid coding schemes, when what many sociologists want to do is to investigate shifts or boundaries of established categories. As Tilly (2002) noted, event cataloguing schemes are, in effect, theories that make assumptions about what kinds of phenomena exist, with templates imposing categories like "protest," "collective violence," "collective action," "conflict," or "instances of contentious claims-making" — which are by no means uncontroversial.

IE systems developed in NLP come with fixed ontologies of relations, events, or entities, and these may often not coincide with how sociologists would like to partition the world in their research.

That said, as is best exemplified by the literature on protest events, there are numerous research scenarios where sociologists may want to assert a more structured coding scheme for the content they study. As we will show below, GenAI offers a highly flexible and easy way for sociologists to design their own IE templates and extract information that is relevant to their research context and questions.

## Studying Obituaries

To examine the potential of GenAI for extracting information, we decided to work on a particular text form: newspaper obituaries. As biographical notes that announce a person's death to a community, obituaries date back to the early days of newspapers. They are a rich source for studying a variety of social sciences questions (Hume 2000; Bytheway and Johnson 1996; Bourdieu 1988). This includes questions concerning the politics of memory and the values and ideologies embedded in whose lives, and which aspects of a life, are considered to be worth commemorating. The content of obituaries provides a rare resource for investigating social norms or social morphology (Fowler 2007; Long 1987; Jallinoja 2011). Which family members are typically constructed as survivors? Which kinds of death are admissible, and which are shrouded in secrecy? Is someone's religious affiliation worthy of mention?

More importantly for this paper, obituaries are a particularly good case for assessing the ability of generative LLMs to accurately extract information in sociological research. For one, obituaries are fundamentally about people and the social context in which they were embedded. They are thus substantially close to textual domains sociologists frequently study. Second, obituaries provide a good tradeoff between highly standardized and highly open textual genres. While there are conventions that govern the tone of the writing and the facts reported, we do find considerable variation. Sometimes, obituaries adopt a rather literary style, while at other times, they are more focused on the bare facts. Besides, obituaries also vary significantly in length and in the order in which they discuss different aspects of the deceased person's life and achievements. Third, and most importantly, obituaries gave us an opportunity to test prompt-based IE on a large and diverse set of pieces of information that are typically scattered across the text. For instance, we extract numeric (e.g., age at death in years), geographic (e.g., the place lived last), and biographical information (e.g., higher education institutions attended). Some variables have a rather limited set of outcomes (e.g., gender), and measuring them can thus also be considered a classification problem.[3] For others, the model is required to extract literal strings of text from the obituaries (e.g., cause of death), while again others require reasoning because the information is not literally stated in the text (e.g., number of children). Some variables, depending on the context, can require the model to summarize information (e.g., we ask for phrases that best summarize the deceased person's occupation).

For our data, we compiled a corpus of all obituaries published in *The New York Times* between 1980 and March 2024. We disregarded paid death notices and commentaries, limiting ourselves to only texts that were written by the *Times'* obituary department, sometimes dubbed the "death beat" (see, e.g., Baranick, Sheeler, and Miller 2005). We also removed duplicate documents and correction statements, leading to a final count of approximately 80,000 obituaries. The average obituary is 706 tokens long, typically including titles with a brief summary of what the person was known for and how old they were, such as "Jane Doe, Art Historian Who Led Restoration of Historic Frescoes, dies at 89."[4] The obituaries typically focus on people who were in the public eye, or who were prominent in a specific field or industry. Indeed, the *Times* described their selection criteria as follows: "people who made a difference on a large stage — people who, we think, will command the broadest interest. If you made news in life, chances are your death is news, too." (New York Times 2022) Their declared aim is to "report deaths and to sum up lives, illuminating why, in our judgment, those lives were significant. The justification for the obituary is in the story it tells." (ibid.)

## From Codebook to Promptbook

### *Defining Variables*

In order to test the potential of generative LLMs for social science-oriented IE tasks, we sought to define a series of variables that are both substantively important but also heterogeneous in terms of their format and with regard to what kinds of capabilities they require on part of the model. In Table 1, we provide an overview of the 12 variables on which we eventually converged.

[3] This is a consequence of us focusing on categorical pieces of information that are, by definition, already linked to one and only one entity (the deceased person). For this paper, we wanted to work with a relatively bounded medium of text that would allow us to test extraction for a large variety of different kinds of information. As we note in our *Discussion* section, IE templates can be designed more complex than we did here. For instance, a more complex example of IE would be to extract all names and corresponding genders from one newspaper issue's entire obituary section.
[4] For this paper, we decided to anonymize and modify text from obituaries. While obituaries are public, we suspect people might object to having their deceased relatives' obituaries used for illustrating a methodological point in a research paper. We also think that this choice makes no substantive difference to our arguments.

The variables are ordered by Task type. This is not a formal typology of IE tasks, but rather our summary of what each task entails. The variables *age in years* and *cause of death*, for instance, typically require identifying a specific number or phrase directly stated in the text and converting them to a uniform format. Similarly, we also extracted the *institutions of higher education attended* by the deceased person and their *religious affiliation*, but these variables also required somewhat more complex inference. For instance, simply extracting all institutions of higher education using named entity recognition is insufficient because many obituaries discuss scholars and where they taught; meanwhile, a person's religious affiliation is often not outright stated as an attribute, but given indirectly through a description of their participation in places of worship (see the notes-column for more details).

Recording the *origin*, that is, the municipality where a person mainly grew up and the *place where they lived last*, can also involve string extraction and inference, given that obituaries typically mention a variety of places, including, for instance, where a person worked or died. Besides, recording this information also often requires exogenous knowledge because municipalities, states, and countries are often not directly mentioned but need to be inferred by the model (e.g., "grew up in the East Village" should be coded as "New York, NY"). Four of our variables require inferring information and assigning a code that is typically not literally stated in the text. In the case of *gender*, this means making a selection between the values "male," "female," and "other," which we made based on pronouns, names, and honorifics (all cases we encountered in the sample used either male or female pronouns). Similarly, we record whether the obituary mentions that the deceased *served in the military*, but this information can be expressed in a variety of ways. The *highest level of education* must be inferred based on different pieces of information. It is worth noting that these three variables are categorical and can thus be treated as classification problems that require a selection among a set of predefined values. Finally, the *number of children* is typically not mentioned in an obituary but must also be inferred based on information that is often scattered across the text.

Beyond these, two of our variables (decedent's *survivors*, decedent's *occupation*) are best described as a summarization task. Most obituaries list *survivors* at the end (e.g., "She is survived by her wife Jane Smith of Tampa, FL; her daughter Mary Johnson of […]"). These sentences, however, can vary considerably in their form. For instance, some survivors may be mentioned in antepositions ("in addition to his wife, Smith is survived by […]") or different forms of enumeration ("among his survivors are his nephews John, Joe, and Jack."). Second, we also code a term or phrase that best summarizes the most important *occupation* of the deceased person. This is not a mere string extraction, as obituaries would often describe a person's professional achievements without using a term that denotes the profession itself. Second, obituaries often mention a variety of positions and job titles, some of which may be peripheral to the person's career (e.g., "at age 13, he worked as a lift boy"), while others may be too specific (e.g., "worked as assistant manager for inventory control"). While there are obituaries that contain appropriate phrases for this variable, more often than not, this task requires summarizing and weighing various pieces of information in the obituary against each other.

An important conceptual difference between IE and discourse-analytic text coding concerns the distinction between the empirical facts of a person's life and the discourse about them represented by obituaries. Our variables ultimately target the latter, not the former. For instance, the absence of a reference to military service does not imply that a person did not serve in the military; and neither does mention of such service technically prove that the person indeed served. Similarly, names, places of origin, and ethnic backgrounds might strongly suggest a particular religious affiliation, but we only code such an affiliation if it is directly indicated in the forms described above. Therefore, the target value for our variable is not the actual religious affiliation of the person, but the one conveyed in obituary discourse. This is perhaps a subtle distinction, but we think it is an important one to keep in mind. Beyond the discussion here, the Notes column in Table 1 contains additional information for each variable, and a detailed overview of all variables is given in the promptbook provided in the Appendix, the development of which we will discuss in the next section.

| **Variable** | **Task type** | **Format** | **Example response** | **Notes** |
|---|---|---|---|---|
| Age in years | String extraction | Integer | "94" | The age is usually stated in the title of an obituary. However, the format of these titles varies. In some cases, the information is more deeply embedded in the text. In a few cases, the information must be inferred based on the birth and publication date. |
| Cause of death | String extraction | Text string | "injuries sustained in a car accident" | Most obituaries state a cause of death but these causes are not standardized. We extract the phrase that specified the cause of death. Usually, these appear in the first few sentences of the obituary but in a few cases, we also saw them occur later in the text. |
| Institutions of higher education attended | String extraction, some inference | Text string | "Rutgers University, University of Michigan" | For this variable, we extract the names of all institutions of higher education that the person attended. This requires string extraction, but can also require inference, given that sometimes such institutions are mentioned but the person didn't actually attend them, or there are statements that implicitly imply attending such an institution (e.g., "He was a quarterback at Michigan.") |

| | | | | |
|---|---|---|---|---|
| Religious affiliation | String extraction, inference | Text string | “catholic” | This variable records the religious affiliation of the deceased person if and only if it was explicitly mentioned. We do not make inferences based on name, origin, or other other categorical attributes of the deceased person. Only a person’s voluntary association with religious organizations or lifestyles (e.g., donated to a specific church or active roles at places of worship) are sufficient to code religion. Applying these criteria, only about one in ten obituaries mention the religious affiliation of the deceased person. |
| Origin | String extraction, inference, exogenous knowledge | Pre-formatted text string | “Queens, NY” | This variable records the municipality where the person grew up in together with a state code or country name. Often, though not always, this place is mentioned somewhere in the text. The variable requires inference, however, because an obituary typically mentions numerous places where the person lived or worked at different times. At times, multiple places are even mentioned in connection to the person’s youth and a selection needs to be made on what is the most significant one. Furthermore, places and states are not always mentioned and need to be inferred by the model based on exogenous knowledge (e.g., “grew up in the East Village” would imply New York, NY). |
| Place lived last | String extraction, inference, exogenous knowledge | Pre-formatted text string | “Paris, France” | Much of what applies to the origin variable also applies here. Places lived last are typically mentioned. When obituaries do not mention such a place but mention where the person died, we record this place. |
| Gender | Inference | Categorical | “female”, “male”, “other” | This categorical variable captures the gender of the deceased person. Measurement requires inference on part of the model but this inference is relatively simple. Effectively, this is a classification problem with three possible outcomes. |

| | | | | |
|---|---|---|---|---|
| Military service | Inference | Binary | “Yes”, “Not mentioned” | The fact that a person served in the military can appear anywhere in the text. It is sometimes stated explicitly (“During WWII, Smith served in the navy […]”) , but at other times only mentioned in passing (“the former platoon leader said …”). Note that an obituary not mentioning that the person served does not necessarily mean that the person actually did not serve. |
| Number of children | Inference | Integer | “4” | This variable requires making an inference about the number of children of the deceased person based on information that can be scattered throughout the text. The number of children is typically not stated literally in the text. Instead, individual children can be mentioned throughout the text. |
| Highest level of education | Inference | Categorical | “Less than high school”, “High school”, “Some college”, “College”, “Masters, PhD, or equivalent”, “not inferable” | This variable requires the model to make an inference about the person’s highest level of education. It is effectively a form of zero-shot classification, for there are only five valid options the model must choose from. |
| Survivors | Summarization | Pre-formatted text string | “1 wife, 2 sons, 2 daughters, 1 sister, 6 grandchildren” | Most obituaries contain sentences that list survivors at the end (e.g., “Besides her husband Dennis of Tampa, FL, Jane Smith is survived by […] ”). We encode this information in a string that contains the family roles of the people mentioned, along with a count. This requires the model to summarize information that is stated in a different format. |

| | | | | |
|---|---|---|---|---|
| Occupation | Summarization | Text string | "politician, lawyer" | In this variable, we record phases that best summarize the occupation of the deceased person. Usually, this is a single word or phrase, but in some instances where a person had multiple significant occupations, we record up to two occupation phrases. Sometimes these phrases are mentioned directly in the text, but often this requires the model to summarize content. For instance, the obituary of a politician may not use the word "politician." Furthermore, this requires distinguishing significant occupations from those that are merely a minor item of the obituary (e.g., "he worked as a liftboy"). |

**Table 1. Variables for Information Extraction.**
*Note: This table gives a summary of the different pieces of information we coded. For more detailed information on the variables, we refer to the promptbook in the Appendix.*

## *Developing a Codebook*

We began our analysis by drawing a stratified random sample of 300 obituaries, ensuring that our data are equally distributed across the 44-year time span. This means we assigned 3,600 unique values for the 12 different variables. We used these cases to develop our variables, come up with clear definitions and coding guidelines, run initial tests of the model, and develop our prompts. We will refer to these data as our *development set*, though we recognize that in classic machine learning, this term typically denotes data used for hyperparameter tuning, and it is debatable to what extent prompt modification is a direct analog of that. Furthermore, note that we chose to structure our paper by first introducing the variables (see previous section), but that some of these variables emerged as we engaged more deeply with the data.

Our coding of the obituaries aligns with the standard procedures for annotation in content analysis and supervised learning projects (for extensive discussions of this, see Krippendorff 2018; Grimmer, Roberts and Stewart 2022, Ch. 18). First, the three authors independently dove into subsets of the data with minimal written rules on how to code each variable. During this process, we would discover cases that were challenging to encode information from based on the instructions we had. This led us to expand our codebook and add new or specify existing rules in the codebook. As readers who have done this kind of coding will know, it usually takes a certain number of cases to establish conventions among coders, even if the challenges that arise are relatively mundane. For instance, if an obituary mentions that someone "took classes" at a college, should this person's highest level of education be coded as "College" or "Some college"? (We decided on the latter). If an obituary states where a person was born but discusses how they spent their youth at a different place, should the first or the second be coded as *origin*? Similarly, when comparing codes for doubly annotated cases, we discussed instances in which we had set diverging values for a variable. Especially for the variables that depend on string extraction, this would require revising our codebook so that we would record the information in the same format. At a certain point, we got the sense that we had sufficient generalizability in the sense that new cases would not take on forms unaccounted for in the codebook, but also reliability in that we would come to the same conclusions.

### *Choosing a Model*

Choosing an appropriate model can be difficult in an increasingly disorienting environment where new models are released on a daily basis and typically advertised with grandiose claims. Additionally, we observe that state-of-the-art models are increasingly available in a multiplicity of variants with different numbers of parameters, levels of quantization, or degrees of fine-tuning, Meta fittingly describing its most recent releases as "the Llama 3 herd of models" (Dubey et al. 2024). A detailed assessment of the tradeoffs of different models is outside of the scope of this paper, and we believe that lasting advice regarding model choice is, in any case, not feasible. That said, three broad considerations guided our selection. All three of these have recently received a more comprehensive treatment by Chae and Davidson (2024), which is why we will keep our discussion relatively short.

First, we argue that social science research should use *open-source models*, that is, models with publicly available and downloadable weights. As has been pointed out by others (e.g., Spirling 2023, Davidson 2024), this is important to ensure reproducibility of the research, as models that are only accessible through an API may be subject to changes in availability, training data, model weights, and other parameters beyond researchers' supervision. Additionally, API-based models would necessitate sharing data with a third party, which may be problematic for researchers who are working with proprietary or sensitive data. Besides, recent developments in large language modeling indicate that the performance gap between open-source models and proprietary models like GPT is closing (Dubey et al. 2024). While there may be scenarios that provide an exception, open-source models on appropriate servers should be the default choice for social science research (for discussion of this, see Palmer, Smith and Spirling 2024).

Second, the kind of prompt-based IE that we do here, but also few- and zero-shot learning in general, work best with models that have been *fine-tuned to follow instructions*. This fine-tuning is typically achieved through a process referred to as reinforcement learning with human feedback (RLHF) in which humans rate the quality of model responses, though in practice, the process also heavily relies on synthetic augmentations of human data (see, e.g., Dubey 2024, pp. 17-18). Model variants that have undergone this process are typically labeled as "-instruct" versions, and popular open-source foundation model families like Llama (Dubey et al. 2024) or Mixtral (Jiang et al. 2024) typically release a base and an -instruct version. For most prompt-based tasks, social scientists are likely to be best served by models fine-tuned to follow instructions.

Third, model choice is limited by the available hardware. We, like most other researchers, are limited in terms of our resources. Additionally, considering ours as a model project that others might follow with a similar setup, it seemed counteractive to us to use a model that many researchers would be unable to run. While the precise computational costs and hardware requirements of running a model depend on more aspects, the two main parameters to consider are model size (i.e., the number of parameters), and the level of quantization. Quantization (Gray and Neuhoff 1998) is a technique that reduces the precision of the model weights by converting them from higher bit depths (e.g., float32) to lower ones (e.g., 4- or 8-bit integers). This considerably lowers the memory demands of running a model and speeds up inference, typically with only minimal loss in performance (for a detailed discussion of quantization techniques and the broader topic of model compression, see Zhu et al. 2024).

With these three considerations in mind, we decided to test two versions of Meta's Llama 3 instruct model, the one with 8 billion parameters (8B) and the one with 70 billion parameters (70B). For both of these, we used quantized versions of these models in GGUF format[5] and ran inference using the llama.cpp library (Gerganov 2024). The 70B variant is about 50 GB in disk size and was run using two NVIDIA V100 PCIe GPUs with 32 GB each. This is hardware that few researchers will have themselves, but that may be accessible to many via their university's High Performance Computing (HPC) resources (which is how we ran our models), or that can be rented from most cloud computing providers. The 8B is about 6 GB in size and was run on a single V100 GPU on our HPC cluster, but could be run on interactive notebook services like Google Colab.

Finally, for all our applications, we follow the precedent set by recent work using GenAI in the social science applications and set the model temperature — a hyperparameter controlling the randomness of a language model's outputs — to 0 (see, e.g., Chae and Davidson 2024, Burnham 2024). This effectively constrains the model to output the most likely tokens. We note, however, that the effects of temperature on the quality of models' inference is an active research area. A recent paper found that for a variety of problem-solving tasks and prompting approaches, varying temperature from 0 to 1 did not have a significant effect on model performance (Renze and Guven 2024).

[5] Specifically, we used the quantized version of Llama 3 70B Instruct made available on the Model Hub at huggingface.co under "Bartowski/Meta-Llama-3-70B-Instruct-Q5_K_M.gguf" and "Bartowski/Meta-Llama-3-8B-Instruct-Q5_K_M.gguf."

## *Developing a Promptbook*

We use the term *promptbook* here in analogy to codebook. By promptbook, we simply mean the complete set of prompts used in a project that builds on prompt-based inference to extract information or label data. In this section, we describe how we developed our promptbook and also discuss some of the difficulties we encountered, which we expect others may also face when following a similar path.

We began developing a promptbook by taking our codebook as the starting point. While it is certainly worth familiarizing oneself with prompting strategies (see, e.g., White et al. 2023), we think that this should typically not be the starting point for social scientific measurement as it risks conceptual slippage. Instead, as we exemplify here, the starting point should be to develop an intersubjectively shared understanding of the quality to be measured, codified in a codebook. As a first step, we therefore simply prompted the model using the codebook we had established to annotate our data. For each variable, we created one prompt based on the codebook instructions with minimal changes.

For instance, we added a sentence at the beginning and at the end of the prompt to introduce the obituary text, and changed the language from asking a coder to "insert" a value to asking the model to "respond" with a value (see Table 2). We also added some formatting-related instructions, to ensure consistency and prevent verbose responses (i.e., to give answers like "[94]" instead of "Certainly! Here is the age: 94"). We also used a system prompt to instruct the model to be rigorous, concise, and to closely adhere to our instructions (see documentation in the Appendix). For simpler variables, such as *age in years* or *military service*, these minimal adjustments improved the accuracy of responses in our development set. However, other variables proved more challenging.

The general procedure we followed is best described as one of iterative optimization. We'll also refer to this as our *development phase* in which we adjust the prompt to improve model performance. During this phase, we worked with the 70B variant of the model. First, we would run a prompt against our data and retrieve a set of values for a given variable. We would then compare these values with the manual labels in our development set. Third, based on this comparison, we made tweaks to the prompt by adding conditions or revising the wording. In making the revisions, we paid special attention to avoiding conceptual "drift," that is, adding language that shifts the meaning of the target concept we intend to measure. We would then repeat this procedure until we either achieved performance deemed satisfactory or couldn't get much improvement through revisions. Through this process, we noticed some general patterns.

| Codebook instructions | Prompt instructions |
|---|---|
| Based on both the title, publication date, and the obituary text, infer the person’s age at death in years and insert it as a numeric value. If age is not inferable, leave this field blank. If the person's age is not inferable, insert 9999. | `Below I will provide an obituary of a deceased person.`<br><br>`Based on both the title, publication date, and the obituary text, infer the person’s age at death in years. You should respond with a numeric value. If the person's age is not inferable, respond with “9999]”.`<br><br>`–`<br>`Please format your response in plain text, inside quotation marks, like this: “<insert your response>”`<br><br>`Here is the obituary date, title, and text: {insert obituary text}` |
| *Does the obituary mention that the person served in the army? If so, put “yes.” Otherwise, put “not mentioned”.*<br><br>*If a person served in a foreign (that is non-US) army, also put “yes.” Advisory roles do not count as having served in the army.* | `Below I will provide an obituary of a deceased person.`<br><br>`Does the obituary mention that the person served in the military? If so, respond with “yes”. Otherwise, respond with “not mentioned”.`<br><br>`If a person served in a foreign (that is non-US) military, also respond with “yes”. Please limit your response to only one of these two codes: "yes", "not mentioned".`<br><br>`Advisory roles do not count as having served in the military.`<br><br>`–`<br>`Please format your response in plain text, inside quotation marks, like this: “<insert your response>”`<br><br>`Here is the obituary date, title, and text: {insert obituary text}` |

**Table 2. Codebook and prompt instruction for age in years and military service.**
*Note: Changes are marked up in gray.*

First, prompts need to be more *explicit* than codebooks for human annotators typically are. Ideally, a codebook should spell out coding instructions in such detail and clarity that two untrained coders could use it to come to the same conclusions about any case. However, in practice, most codebooks are likely to rely on a certain level of implicit understanding of what a task is about and what ends it serves. This is especially likely if annotations are not crowd-sourced but done not by a group of researchers who interact with each other, as will be true for most projects. For instance, when coding the *highest level of education* variable, we had an unarticulated, shared understanding that we would code this variable based on what the obituary says about a person’s educational trajectory. Yet when using our codebook instructions for prompting, we quickly noticed that the model drew inferences based on other pieces of information. People with certain occupations, or who were highly-achieving professionally, were often coded as having a college education, even if no such education was explicitly mentioned in the text. In retrospect, the wording in our codebook ("Based on the entire obituary, to the best of your ability, infer [...]") does indeed not preclude coding the variable in this way and can perhaps even be read as encouraging such inference. Meanwhile, by "infer" (as far as we can realistically recollect), we had primarily meant to draw conclusions about which of the potentially many educational degrees was the highest. This then led us to revise the language in the prompt and to add an explicit instruction for the model not to make such inferences (see Table 3). RLHF is, of course, precisely about inducing (unstated) anticipations about user preferences into models, but our experience suggests that models need things to be spelled out that human coders typically do not. On the upside, as a third party devoid of research-specific contextual knowledge, they can help researchers identify unarticulated assumptions that perhaps should have been spelled out in the codebook in the first place.

| **Codebook instructions** | **Prompt instructions** |
|---|---|
| *Based on the entire obituary, to the best of your ability, infer and record the highest level of education of the deceased person as one of the following options:*<br>*Less than high school, High school, Some college, College, Masters, PhD, or equivalent, Not inferable*<br><br>[abbreviated] | `Below I will provide an obituary of a deceased person.`<br><br>`Based on what the text says about this person's education, record the highest level of education of the deceased person as only one of the following codes:`<br>`"Less than high school", "High school", "Some college", "College", "Masters, PhD, or equivalent", "not inferable"`<br><br>`When giving your response, consider the following rules:`<br><br>`1) Generally, do not infer this person's level of education from their occupation without explicit statements in the text about the degrees they obtained. The only exceptions to this are people who teach at universities, people who practice law, or those who practice medicine. In those cases, respond with "Masters, PhD, or equivalent".`<br><br>`[abbreviated]` |

**Table 3. Codebook and prompt instruction for the highest level of education.**
*Note: Changes are marked up in gray. Texts were abbreviated and the full prompt is provided in the Appendix.*

We also noticed that models can be stubbornly prone to specific, often simple kinds of errors. For instance, when coding *institutions of higher education* that the deceased person attended, the model would regularly include educational institutions that were not postsecondary like high schools, institutions that were mentioned but attended by others, and institutions where the person taught as a professor but that they did not attend as a student. We have no theory as to why the model would make these particular errors, but we can say that nothing in our initial codebook-based prompt can be read as encouraging or permitting these kinds of behaviors. We addressed this problem by adding extensive explicit instructions asking the model not to make a particular kind of error (see Table 4). This definitely helped improve on some cases, but it also didn't solve the issue, leading us to our next observation.

Researchers who define their own IE templates will notice that some tasks, and not always those one would expect, are simply hard for a model. In our case, even after adding instructions, the model would regularly respond with educational institutions that it shouldn't respond with. To tackle this, we asked the model to provide not just the value but to first provide relevant evidence from the obituary and then a rationale for its eventual response (see Table 4). Such strategies, discussed under the label *chain-of-thought prompting*, and have been shown to have potential for improving response behavior (Wei et al. 2022), including in social science applications (see, e.g., Dunivin 2024, Underwood 2023). The general ideal behind chain-of-thought prompting is that the model generates intermediate reasoning steps (e.g., selecting relevant evidence) that allow it to break down complex problems into smaller, more manageable parts before arriving at the final response. As we will show in the next section, this technique indeed elevated the model's performance on some tasks, but not on others.

In this context, we note two challenges we encountered when developing our prompts. Firstly, prompts can be tweaked in nearly infinite ways. In classic machine learning, there is typically a limited, if large, number of possible hyperparameter configurations. Various techniques exist for systematically searching this space for ideal parameter combinations. Prompt tuning (Li and Liang 2021; Lester, Al-Rfou, and Constant 2021) — the unsupervised optimization of prompts for specific tasks — may be a promising equivalent for prompt-based IE. When engineering prompts manually as we did here, however, it is difficult to navigate the space of possible and promising alterations, especially while guarding against conceptual drift. Adding to that, we caution against a temptation to over-engineer prompts under the assumption that a model can handle an inference task if we just prompt it well enough, as we do with human coding assistants. For example, even after we added explicit instructions not to record the university that someone worked at for the variable described above, the model would still make this error. Despite our best efforts to revise our language and add additional reminders, we eventually concluded that we couldn't reliably prevent the model from making this mistake. Although we cannot exclude the possibility that some prompt does exist to resolve the issue, our experience showcases the challenges to reliably preventing model errors.

<table>
<tr><th>Codebook instructions</th><th>Prompt instructions</th></tr>
<tr><td>Record all institutions of higher education that were attended by the person (i.e., universities and colleges, or graduate & professional schools). These should be recorded even if the person did not complete their degree.<br><br>Institutions should be typed exactly as they appear in the text.</td><td>Below I will provide an obituary of a deceased person.<br><br>Record all institutions of higher education that the person obtained a degree from (i.e., universities, colleges, or graduate & professional schools), exactly as written in the text. If the text indicates that this person attended some institution as a student, but did not complete their degree, record this institution as well. When giving your response, consider the following rules:<br><br>1) Do not include high schools or college preparatory schools.<br>2) Do not include institutions that the person’s friends, family, coworkers or partners attended, unless the deceased person also attended them.<br>3) Obituaries may describe decedents who were employed at academic institutions, such as instructors, scientists, university administrators and coaches. You must distinguish higher education institutions that this person studied at from those that this person worked at. Only institutions where the person studied should be considered in your response. Do not record higher education institutions only because the person worked, taught, or held a job there. For example, if the text says “after transferring from University 1 to study mathematics at University 2, he eventually got a master's degree from University 3. He became a head coach at University 4 and taught sports science at University 5”, your response should only include Universities 1, 2 and 3, but not University 4.<br><br>If the text does not mention any institutions of higher education that the person attended, simply respond with “none”. Otherwise, your answer should include a rationale, as well as quotes from the text as evidence. Your response should be formatted as a JSON file that follows this template:<br><br>{<br>  "evidence": ["<insert quote 1>", "<insert quote n>"],<br>  "rationale": "<insert your rationale>",<br>  "higher_education_institutions": ["<insert institution 1>", "<insert institution 2>", "<insert institution 3>"],<br><br>}<br><br>Here is the obituary date, title, and text: {insert obituary text}</td></tr>
</table>

**Table 4. Codebook and prompt instruction for attended institutions of higher education.**
*Note: Changes are marked up in gray.*

This then points to a more general dilemma: how can researchers develop a sense of whether their prompts are "good enough," or when they reach the model's performance limit such that further tweaking will not improve performance? Put differently, how do we know when to stop prompt engineering? We cannot offer a definitive solution, but we will describe the heuristic we followed in the hope that it proves useful. Generally, when evaluating model responses against our development set, we noticed two types of model errors: systematic and unsystematic ones. We focused our attention on the former. For instance, when coding military service, we noticed that the model would, strangely, code people who served in the Navy or the Air Force as "not mentioned." Upon checking our prompt and codebook, we realized that we had used the term "army" with the colloquial understanding that it refers to the military as a whole. Substituting it for "military" in our prompt improved the model's performance significantly. We stopped prompt engineering once the model's remaining mistakes seemed unsystematic and could not be directly addressed by further tweaking the prompt, or when systematic errors remained, but these could not be fixed by introducing explicit instructions to avoid them. We recognize, of course, that whether or not something appears "systematic" under scrutiny depends on its prevalence, sample size, and the researchers' capacity to recognize this systematicity. The concrete implications of our heuristic will thus depend on the context and the researcher.

# Evaluation

## *Test Set and Model Scenarios*

All evaluations presented in this section are made on a new, double-coded stratified random sample of 200 cases. We refer to this as our *test set*. We also blinded the obituaries to prevent the model from responding based on exogenous knowledge it might have learned about the deceased people from its training data. [6] We did this by replacing all first and last names with generic names. Men's first name was replaced with "John," and subsequent middle names were replaced with a single middle name, "Michael." Women's respective names became "Jane" and "Mary." The last names were replaced with "Smith" and "Johnson."

We assess the performance on our twelve IE tasks for a set of different prompting strategies. Specifically, for all our variables, we examine the effect of *chain-of-thought prompting*,

[6] It is worth noting that we cannot exclude the possibility that the model might still be able to infer the true identities of the people based on other pieces of information in the obituary. There is unfortunately no way to fully blind these data, but this will likely also be true for many other social science applications. Still, for some research applications, the model's capacity to draw on exogenous knowledge can be advantageous. This is somewhat exemplified in our variables about *origin* and *last place lived* where the model drew from general geographical knowledge. However, it is arguably not true for our tasks at large, given that, as we stated above, our variables are meant to encode aspects of the discourse represented in obituaries, and not the actual lives of people portrayed in them.

as opposed to just asking for a response directly. We also examine whether asking the model to structure its response in *JSON* format or just asking it for plain text makes a difference. Beyond this, our discussion of the promptbook so far has focused on zero-shot prompting, that is, we ask the model for a response to a single obituary alone, without giving it any prior examples. Going beyond this, we also implemented a one-shot prompting scenario, giving the model one example and one simulated correct answer, then providing the actual obituary to label (see Appendix D for details).[7] The idea behind this strategy is that examples allow the model to recognize expected behavior. The evidence on whether this technique improves model performance is inconclusive, suggesting that it is effective for some tasks but not for others (Brown et al. 2020, Chae and Davidson 2024).[8]

The results we present below are based on 19,200 distinct model executions. That is, for each of the 200 cases in our test set, we prompted the model 96 times, covering twelve variables, four prompting strategies, and two differently sized models. Below, we provide a curated summary of our results, focusing on the key patterns we consider to be most relevant. The full results are presented in Table 5 (70B model) and Table 6 (8B model).

### *Results by IE Tasks*

The most important question we want to answer is whether the generative large language models can accurately extract the kinds of information we discussed above. Therefore, before getting to the differences between different models and prompting strategies, we first focus our discussion on the results of our largest and best-performing model (Llama 70B Instruct), broken down by variable. Figure 2 shows the accuracy of this model for each of our tasks (red). *Accuracy* (Acc.) is the share of cases for which the model's response corresponded to the value we assigned when coding the test set. When assessing this, we used minimal post-processing for the model responses as detailed in Appendix E. The accuracy scores in Figure 2 are averages across the four different prompting strategies (see Table 5 description for details). We emphasize that researchers who want to extract information for downstream tasks will have to conduct their own evaluations, and that what counts as sufficient performance will depend on aspects like sample size and specific analytic goals.

---

[7] Not least because obituaries are quite long, we don't explore few-shot prompting (i.e., giving the model more than one example) and limit ourselves to exploring a one-shot prompting.

[8] Considering the high number of tasks we evaluate here, we limit ourselves to four scenarios instead of testing all eight possible combinations of these choices. Our default scenario asks for JSON-formatted responses, uses chain-of-thought prompting, and is structured as a 0-shot. The other three scenarios respectively alter one parameter in this default scenario.

The 70B model achieved almost perfect accuracy for *gender* (1), *age in years* (.99), *military service* (.99), and the *number of children* (.98).[9] The first two variables may be seen as relatively easy targets: gender can be identified by linguistic features such as the prevalence of certain honorifics ("Mr.," "Mrs.," "Ms.," etc.) or gendered pronouns ("she," "her," "he," "him," "his") that most conventional machine learning classifiers could have picked up on. Age, as we discussed above, is often mentioned directly in the title or early in the text in a relatively standardized wording (e.g., "died at the age of [...]"). Both military service and gender are binary (in our test set) and have roughly similar imbalances (with 22.5% women and 19% people with stated military service). But whereas gender typically permeates the whole obituary via pronouns and names, military service is typically mentioned in passing somewhere deeply embedded in the text. It is impressive that the model succeeds at picking this up, for conventional supervised learning methods can struggle with this kind of sparse-signal information. Finally, unlike the previous variables, extracting the number of children required identifying information that is not literally stated in the text. The model's strong performance on this task highlights the effectiveness of our prompt-based approach in tackling an IE task that involves summarizing and inferring information from relatively long texts.

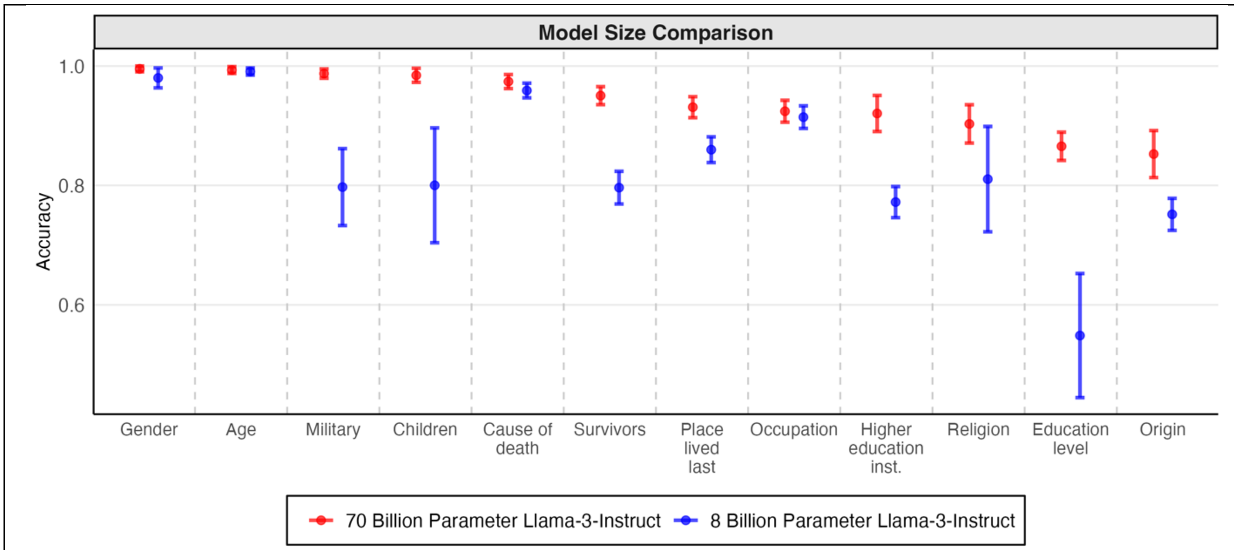


**Figure 2. Accuracy Across Information Extraction Tasks**
*Note: The figure visualizes the accuracy of the 70B and 8B parameter models across our twelve IE tasks. These estimates and the corresponding 95% confidence intervals were obtained by fitting independent random-effects meta-analysis models for each task and model that treat the results for each of the four prompting strategies as separate studies (n = 200, respectively). In this way, each result is based on 800 model executions. All accuracies are contained in Tables 5 and 6. For the six studies with perfect accuracy, we subtracted a small constant to integrate them into our modeling framework, replacing accuracy with* $n/n+1$.

[9] The few errors that there were for *age* stemmed from cases where the information was not present, but rather than stating this, the model would make dubious inferences.

*Tasks are ordered along the x-axis by average accuracy for the 70B model.*

Our models also did well when it came to extracting the *cause of death* (acc. of .97). This shows that the 70B model can successfully identify and extract relevant text strings. The variable that records *survivors* is best described as a summarization task. Survivors are usually mentioned in a subsection at the end of an obituary, but as we discussed above, the lexico-syntactic heterogeneity in these expressions makes the extraction of survivors non-trivial. Like for the *number of children*, the model must not just detect references to surviving kin but also enumerate them and distinguish them from kin mentioned in passing. The fact that the model reached an accuracy of .95 on this task attests to its ability to perform several inferences at a time. Besides, incorrect responses were often partially correct, missing only one or two survivors; we report these results in Table 5, which average .97.

| **Model scenario** | | | | | |
|---|---|---|---|---|---|
| Size | | 70B | 70B | 70B | 70B |
| Requested JSON format | | Yes | No | Yes | Yes |
| Chain-of-thought applied | | Yes | Yes | No | Yes |
| Shots | | 0 | 0 | 0 | 1 |
| **Task** | **Measure** | | | | |
| Gender (categorical) | Acc | **1** | **1** | **1** | **1** |
| | Acc. CIs | [1, 1] | [1, 1] | [1, 1] | [.99, 1] |
| | F1 | 1 | 1 | 1 | .67 |
| Age in years (integer) | Acc. | .99 | .99 | .99 | **1** |
| | Acc. CIs | [.98, 1] | [.98, 1] | [.98, 1] | [.99, 1] |
| Military (binary) | Acc. | **.99** | **.99** | .98 | .98 |
| | Acc. CIs | [.98, 1] | [.98, 1] | [.96, 1] | [.97, 1] |
| | F1 | .98 | .98 | .97 | .98 |
| Children (integer) | Acc. | .97 | .98 | **1** | .98 |
| | Acc. CIs | [.95, .99] | [.95, 1] | [.99, 1] | [.97, 1] |
| Cause of death (text string) | Acc. | .96 | .96 | **.98** | .97 |
| | Acc. CIs | [.94, .99] | [.94, .99] | [.97, 1] | [.95, .99] |
| Survivors (pre-formatted text string) | Acc. | .93 | **.96** | **.96** | .94 |
| | Acc. CIs | [.89, .97] | [.93, .99] | [.93, .99] | [.91, .97] |
| | Acc. partial | .96 | .97 | .98 | .96 |
| Place lived last (pre-formatted text string) | Acc. | .92 | .93 | **.94** | **.94** |
| | Acc. CIs | [.88, .95] | [.89, .97] | [.91, .97] | [.90, .97] |
| Occupation (text string) | Acc. | .92 | **.93** | .92 | **.93** |
| | Acc. CIs | [.88, .95] | [.89, .97] | [.88, .96] | [.89, .97] |
| | Acc. partial | .98 | .98 | .99 | .97 |
| Institutions of higher | Acc. | **.96** | .90 | .88 | .92 |
| | Acc. CIs | [.93, .98] | [.86, .95] | [.84, .93] | [.89, .96] |

| | | | | | |
|---|---|---|---|---|---|
| education attended (text string) | Acc. partial | .97 | .94 | .92 | .95 |
| Religious affiliation (text string) | Acc. | .86 | .90 | **.94** | .92 |
| | Acc. CIs | [.81, .9] | [.85, .94] | [.90, .97] | [.88, .95] |
| | Acc. partial | .86 | .90 | .94 | .92 |
| Highest level of education (categorical) | Acc. | **.88** | .86 | .86 | .86 |
| | Acc. CIs | [.83, .93] | [.82, .91] | [.81, .91] | [.81, .90] |
| | F1 | .82 | .83 | .73 | .73 |
| Origin (pre-formatted text string) | Acc. | .83 | .82 | **.90** | .84 |
| | Acc. CIs | [.78, .88] | [.77, .88] | [.86, .94] | [.79, .89] |

**Table 5. Accuracy of Information Extraction for Scenarios with 70B Model.** Note: The table reports the accuracy of information extraction for the four scenarios with the 70B Llama instruct model as well as the baselines, where applicable. Obituaries were blinded by replacing the first, middle, and last names of the person discussed in the obituary. This includes replacing names of their partners, heirs, siblings, and other people mentioned if these had the same names. Additionally, we report the macro F1 scores for categorical outcomes. For some tasks, we provide a measure of partial accuracy (Acc. partial), which includes cases where the model's response partially matches the human response. For instance, for extracting survivors, incorrect responses were typically at least partially correct but missing one survivor, sometimes two. Partial accuracy, by definition, is higher than our main accuracy measure. All results are based on a time stratified random sample of 200 obituaries. To reflect the uncertainty of our accuracy values, we use normal approximation to compute 95% confidence intervals (CIs) for each accuracy via $\text{Acc.} \pm z\sqrt{\frac{\text{Acc.}(1-\text{Acc.})}{200}}$. Note that this approach can lead to unrealistic values (e.g., an upper bound of 1) for those tasks with perfect or near perfect accuracy in our sample.

The model performed slightly worse on extracting geographic information: *place lived last* (acc. of .93) and *origin* (.85). These tasks require string extraction, inference, and, at times, exogenous knowledge. In particular, identifying a person's *origin* can require piecing together and weighing scattered information (e.g., school attendance, parent's place of work, events that marked the decedent's childhood, etc.). For both tasks, the errors primarily stemmed from ambiguity in the texts. Also, to the model's credit, it was able to find relevant information sprinkled in various places of the obituaries, for example, by inferring the last place of residence via the location where the decedent's partner lived.

When asked to summarize the main professional activities (*occupation*), the model gave an identical or highly similar answer to human coders in most of the cases (.93. By highly similar, we mean minor variations in wording. While one could use an exact string matching criterion (which would imply lower accuracy), this seemed to us to go against the intent of the task. The partial accuracy for this variable (see Table 5), suggests that for another 5% of the cases, the agreement was at least partial: the model aptly identified one profession but not the other(s). It is worth noting that obituaries published in the Times typically give a great deal of attention to a person's professional achievement, so the error rate might have been higher had this not been the case.

Two tasks that the model struggled with more are *institutions of higher education attended* (acc. of .92) and *highest level of education* (acc. of .87 to .9; average macro-F1 of .78). Concerning the former, we have already indicated that we could not reliably prevent the model from listing institutions that were mentioned for reasons other than the person having attended them, such as the person having taught there. Concerning the latter, the model was generally too strict in recognizing academic achievements: phrases stating that a person "attended" or "went to" a specific university are commonly understood as indicative of degree completion and were treated by us as sufficient for assigning the value "College." Despite our instructions for the model to behave in the same way, however, it would regularly assign the code "Some college." Other errors came from ambiguities regarding the level of an institution, mostly between college and advanced degrees.

Finally, the results for *religious affiliation* merit special attention. On the surface, accuracy (.9) doesn't look too bad (though this is in good part driven by 88% of cases not mentioning the person's religion). However, what is noteworthy is the nature of the errors the model made, which we had already noticed when working with our development set. Despite our best efforts, we could not prevent the model from imputing religious affiliations from information that was not explicitly about religious practice, even though our prompt instructed the model not to infer religion based on identity markers including nationality, name, origin, and such. This indicates a real limitation and risk of using prompt-based IE in social science research, the implications of which we will discuss in more detail below.

### *Results by Model Size*

As we mentioned above, running a 70-billion-parameter model and analyzing large corpora with it involves significant computational and logistical costs, which may be prohibitive for many sociologists. It is, therefore, worth testing whether a smaller model like the 8 billion parameter Llama 3 Instruct can handle our IE tasks similarly well. In Table 6, we report the performance of our 8 billion parameter model. Previously mentioned Figure 2 also contains the average model performances and for the smaller model (blue).

Generally, we find that the smaller model performs worse for all tasks and scenarios. However, there is some interesting heterogeneity. For *gender*[10] (acc. of .97), *age* (.99), *cause of death* (.96), and *occupation* (.91), extraction is almost as accurate as for the 70B model. However, for *military* and *survivors*, where the 70B model had accuracies of .95 or higher, performance dropped sharply (-.20, -.16, and -.15, respectively). We also see declines concerning *place lived last* (- .14), *origin* (- .07), *institutions of higher education* (-.10), and *religion* (- .11). Finally, the poorest performance of the 8B model concerns *level of education* (acc. of .56) where the average accuracy dropped by .30.

| **Model scenario** | | | | | | |
|---|---|---|---|---|---|---|
| Size | | 8B | 8B | 8B | 8B | 8B |
| Requested JSON format | | Yes | No | Yes | Yes | Yes |
| Chain-of-thought applied | | Yes | Yes | No | Yes | No |
| Shots | | 0 | 0 | 0 | 1 | 0 |
| System prompt given | | Yes | Yes | Yes | Yes | No |
| Task | Measure | | | | | |
| Gender (categorical) | Acc | .96 | .96 | **1** | .98 | **1** |
| | Acc. CIs | [.93, .98] | [.93, .98] | [1, 1] | [.96, 1] | [1, 1] |
| | F1 | .65 | .65 | 1 | .66 | 1 |
| Age in years (integer) | Acc. | .98 | .98 | .99 | **1** | .99 |
| | Acc. CIs | [.97, 1] | [.97, 1] | [.98, 1] | [1, 1] | [.98, 1] |
| Military (binary) | Acc. | .76 | .68 | **.86** | .85 | .82 |
| | Acc. CIs | [.70, .82] | [.62, .74] | [.81, .91] | [.80, .90] | [.77, .88] |
| | F1 | .72 | .65 | .80 | .81 | .78 |
| Children (integer) | Acc. | .86 | **.88** | **.88** | .64 | .72 |
| | Acc. CIs | [.81, .91] | [.84, .93] | [.84, .93] | [.58, .71] | [.65, .78] |
| Cause of death (text string) | Acc. | .95 | .95 | **.96** | .96 | **.96** |
| | Acc. CIs | [.92, .98] | [.92, .98] | [.94, .99] | [.93, .99] | [.94, .99] |
| Survivors (pre-formatted text string) | Acc. | .8 | .78 | .78 | **.84** | .76 |
| | Acc. CIs | [.74, .86] | [.73, .84] | [.73, .84] | [.79, .89] | [.70, .82] |
| | Acc. partial | .91 | .88 | .90 | .92 | .92 |
| Place lived last (pre-formatted text string) | Acc. | .84 | .86 | .88 | **.87** | .83 |
| | Acc. CIs | [.79, .90] | [.82, .91] | [.83, .93] | [.82, .92] | [.78, .88] |
| Occupation (text string) | Acc. | .88 | .90 | **.94** | .91 | .93 |
| | Acc. CIs | [.83, .93] | [.86, .94] | [.90, .97] | [.87, .95] | [.89, .97] |
| | Acc. partial | .98 | .96 | .98 | .97 | .98 |

[10] It might surprise that the model would miss cases on a seemingly easy task like gender. Our prompt included the categories "male," "female," and "other," while, at least based on pronouns, all cases in our test set could be classified as either "male" or "female." The errors for gender almost exclusively stemmed from the model being reluctant to assign either "male" or "female" and instead choosing "other." It seems likely that accuracy in our test set would have been higher had we used a prompt with binary options, but this would have come at the cost of excluding an option for non-binary people.

| | | | | | | |
|---|---|---|---|---|---|---|
| Institutions of higher education attended (text string) | Acc. | .74 | .73 | .79 | .79 | **.80** |
| | Acc. CIs | [.68, .81] | [.67, .79] | [.73, .85] | [.73, .85] | [.74, .85] |
| | Acc. partial | .80 | .80 | .86 | .85 | .88 |
| Religious affiliation (text string) | Acc. | .72 | .71 | **.94** | .81 | .86 |
| | Acc. CIs | [.65, .78] | [.65, .77] | [.91, .98] | [.76, .86] | [.81, .91] |
| | Acc. partial | .72 | .72 | .95 | .81 | .87 |
| Highest level of education (categorical) | Acc. | .58 | .56 | .40 | **.72** | .50 |
| | Acc. CIs | [.51, .64] | [.49, .62] | [.33, .47] | [.66, .78] | [.43, .56] |
| | F1 | .41 | .42 | .25 | .58 | .35 |
| Origin (pre-formatted text string) | Acc. | .72 | **.76** | **.76** | .74 | **.76** |
| | Acc. CIs | [.66, .79] | [.71, .82] | [.70, .82] | [.68, .81] | [.70, .82] |

**Table 6. Accuracy of Information Extraction for Scenarios with 8B Model.**
*Note: The table reports the accuracy of information extraction for the four scenarios with the 8B Llama instruct model as well as the baselines, where applicable. For details, see the description of Table 5. Additionally, on the suggestion of one reviewer, we tested one strategy where we dropped the system prompt.*

Beyond this, we note that the 8B model had more difficulties adhering to the requested output format. First, we noticed that in a few cases, the model would be verbose by adding statements like "Here is the response," before giving the actual response. In Appendix E, we provide a systematic analysis of this behavior, showing that for the 8B model this occurs in 0 to 4% of the cases (depending on the variable), while for the 70B model, we only found .1% such cases. Second, while the 8B model also responded with JSON style answers when prompted to do so, unlike for the 70B model, these frequently did not strictly adhere to JSON format (leaving out, for instance, quotation marks around "rationale" items). While it was straightforward to extract the model's responses, our experience suggests that researchers working with the 8B model cannot expect to reliably receive JSON-formatted responses.

Overall, these results lead us to draw a mixed conclusion regarding model size. For some tasks, the smaller model reached similarly high accuracy as the 70B, which would likely be sufficient for many downstream tasks. Especially with large corpora and many cases to annotate, this suggests the potential to save a substantial portion of the computational costs. On the other hand, our analysis also shows that for some tasks, the smaller model fell considerably short of the larger one. One caveat here is that we worked with the 70B model when we developed our prompts, so we can't exclude the possibility that some of the differences we find between models are a consequence of the specific prompts we used. To the extent that performance differences are not conditional on prompts, however, our results suggest that getting good performance on some tasks may indeed require using a larger model. What distinguishes such tasks from those where we saw few or no differences? It appears that the smallest performance differences concerned information that is highly present throughout the obituaries (*gender* and *occupation*) or reliably stated in more or less formulaic form right at the beginning (*age*, and *cause of death*). Meanwhile, some of the most significant performance drops concerned information that is sparse and akin to a needle in a haystack (*military*, *education level*) and/or that requires numeric inference (*number of children*, *survivors, education level*). We will reflect on this observation in our discussion below.

### *Prompting Strategies*

As indicated above, we also tested four different prompting strategies.[11] Our main observation is that none of them performs universally better across the tasks. This holds true for both the 70B model and the 8B model. Especially for the tasks with the highest performance, approaches fared almost equally well, and the differences typically rest on only a few cases. For the tasks where the models performed worse, the variation between approaches is larger, but at least some of this heterogeneity is likely to be driven by chance. In Figure 3, we present accuracy estimates for each strategy for both models. These estimates and the corresponding 95% confidence intervals were obtained by fitting eight independent random-effects meta-analysis models that treat the results for each of the 12 tasks as separate studies. Overall, the differences between the strategies are quite modest.

Even so, one approach stands out slightly: the scenario without chain-of-thought prompting generally performed best, independent of model size. For both the 70B and the 8B models, avoiding this technique led to the best accuracy for 8 of our 12 variables. A qualitative inspection of the results suggests that initiating a chain-of-thought can make models reluctant to state that the information is absent. Instead, models select vaguely relevant textual evidence and proceed to make rather creative interpretations (this was especially the case for *origin*). This aligns with prior findings that generative LLMs have a predisposition to give false positives and to hallucinate when information simply is not present (Ma et al. 2023). On the other hand, for at least one task (identifying *institutions of higher education*) the model performed markedly better with this mode of prompting. Taking up our prior discussion of this variable, this might suggest that chain-of-thought prompting is a good strategy for forcing the model to comply with explicitly stated rules if it otherwise doesn't. However, our results suggest that it is by no means a universally superior strategy, which aligns with some recent findings by Burnham (2024, p. 11).

[11] Additionally, on the suggestion of one reviewer, we tested whether removing custom system prompts can elevate smaller models' performance. The 8B Llama Instruct model's capacity to tackle our IE tasks was not impacted by this (see Table 6).

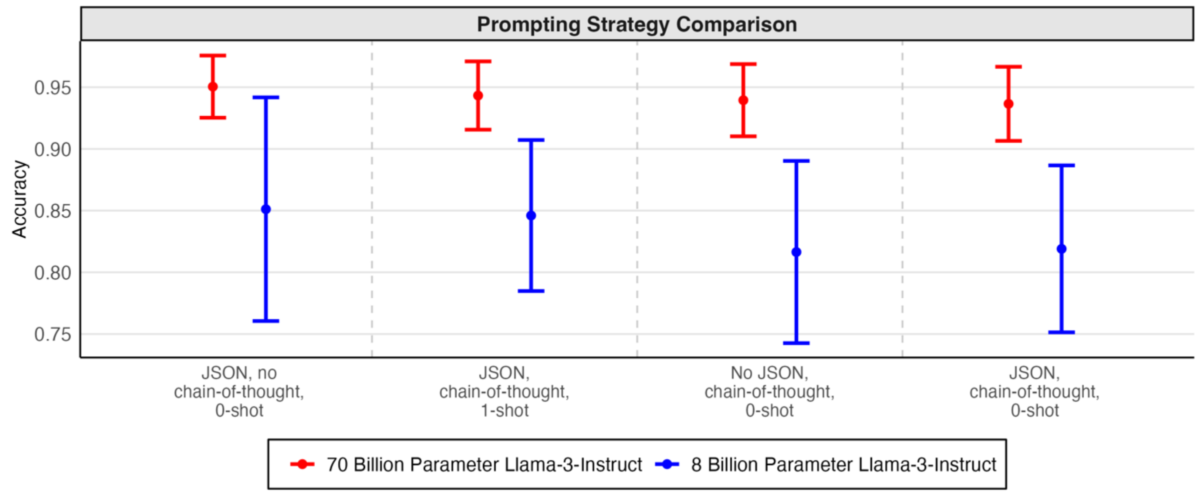


**Figure 3. Comparison Between Prompting Strategies**
*Note:* The figure presents accuracy estimates for each strategy for both models. These estimates and the corresponding 95% confidence intervals were obtained by fitting eight independent random-effects meta-analysis models that treat the results for each of the 12 tasks as separate studies (n = 200, respectively). In this way, each result is based on 2,400 model executions.

## *Baseline Approaches*

Overall, our results point to GenAI's potential for sociological inquiry, showing that especially the 70B, but also the 8B model can serve as an accessible tool for IE. However, at least for some of the tasks we discussed above, there is reason to suspect that simpler, more conventional approaches could tackle them reasonably well. In addition to the different models and prompting strategies described, we created a set of *baseline* approaches that rely on more conventional methods for a subset of our tasks. Our findings are summarized in Table 7.

First, *gender*, *military*, and the *highest level of education* have a fixed number of outcomes and can be considered conventional classification problems. We implement two alternative approaches to tackle this task. First, fine-tuning smaller language models has been shown to be a potent strategy for classification problems, even with small numbers of annotated cases (see, e.g. Bonikowski, Luo, and Stuhler 2022; Do, Ollion, and Sheng 2022; Chae and Davidson 2024). We fine-tune Longformer models (Beltagy, Peters, and Cohan 2020). Second, as an even simpler approach, we implemented Naïve Bayes classifiers that uses the raw document-term matrix. Both approaches are evaluated on our test set via 10-fold cross-validation (see Appendix C for details). We find that for *gender*, the two conventional classifiers perform just as good as the 70B model and even slightly better than the 8B model. However, for *military* and especially for *level of education*, both baseline models did markedly worse than our 70B model. It seems likely that both baselines would improve with more labeled data. Overall, our results suggest that more conventional classification approaches can be an economical alternative to prompt-based extraction if the information is highly present in the text. For sparse-signal information that is more akin to a needle in a haystack, like *military service or details about education*, prompting may be preferable.

<table>
<tr><th colspan="2" rowspan="2"></th><th rowspan="2">70B Model</th><th rowspan="2">8B Model</th><th colspan="2">Baseline</th></tr>
<tr><th>Longformer</th><th>Naïve Bayes</th></tr>
<tr><td rowspan="2">Gender (categorical)</td><td>Acc.</td><td>1</td><td>.97</td><td>.99</td><td>.98</td></tr>
<tr><td>F1</td><td>.92</td><td>.79</td><td>.99</td><td>.97</td></tr>
<tr><td rowspan="2">Military (binary)</td><td>Acc.</td><td>.99</td><td>.8</td><td>.83</td><td>.81</td></tr>
<tr><td>F1</td><td>.98</td><td>.75</td><td>.72</td><td>.55</td></tr>
<tr><td rowspan="2">Highest level of education (categorical)</td><td>Acc.</td><td>.87</td><td>.55</td><td>.5</td><td>.51</td></tr>
<tr><td>F1</td><td>.78</td><td>.40</td><td>.29</td><td>.29</td></tr>
<tr><td>Age in years (integer)</td><td>Acc.</td><td>.99</td><td>.99</td><td colspan="2">.95</td></tr>
<tr><td>Cause of death (text string)</td><td>Acc.</td><td>.97</td><td>.96</td><td colspan="2">.76</td></tr>
</table>

**Table 7. Evaluation of Baseline Approaches versus Generative Models**

*Note: The table shows the average accuracy and macro-F1 scores for our baseline approaches. For direct comparison, we also show the average accuracy and F1-scores averaged across prompting strategies for the 70B and the 8B model.*

Beyond this, we created regular expressions to extract the *age* of the deceased person. When working with our development set, we noticed that age is usually stated at the beginning of obituaries and only rarely requires inference. While we initially tried more complex string-matching approaches, we eventually realized that we could get surprisingly reliable results by simply extracting the first two- or three-digit number that appears in an obituary. With this approach we reach an accuracy of .95, which is only slightly worse than both generative models (.99).

Finally, we noticed that the language around *causes of death* was highly patterned. We systematically looked at obituaries that mention causes of death in our development set and built a system of regular expressions that extract phrases that follow things like “died of,” “died from,” “the cause was,” and so forth (see Appendix C for details). We created this system using only our development set. When running this against out test set, we reach an accuracy of .76. Though we spent considerable time developing this baseline, we note that the approach could be improved by examining more cases and expanding the set of indicative phrases. Nonetheless, this result suggests that even for information embedded in a highly patterned lexical environment, achieving comparable extraction performance as a prompt-based approach might be hard.

Other variables did not seem particularly amenable to this kind of approach as they required inference or the language around the relevant pieces of information was more varied (e.g., occupation). This does not mean that one could not build systems for measuring them that don’t depend on GenAI, say, by extracting noun phrases and building. The baselines aimed to test whether, for at least some IE tasks, researchers can get similar results with simpler and more economical tools. This is indeed the case, which we'll reflect on in our summary.

## Summary and Guiding Thoughts

In this section, we summarize our findings and discuss their practical implications for those who consider extracting information from texts in a similar way. Overall, we consider our results encouraging. They demonstrate that two moderately-sized open-source generative LLMs that scholars can run locally (i.e., not through an API) successfully handled relatively complex inference tasks when instructed well with prompts.

How can researchers know whether their task can be successfully tackled with this approach? The short answer is that they can't know a priori, but our results provide some useful insights. We found that the 70B model had nearly perfect accuracy for extracting information that is more or less explicitly stated in the text (*gender*, *age*, *military service*, *cause of death*). It also did well for variables that require precise numeric inference (*number of children*, *survivors*). Meanwhile, the model's ability was somewhat lower when it came to extracting information that required what one might best describe as interpretive competence (*education level*, *origin*, *religion*): despite our instructions, the model would treat phrases such as that someone "went to" a university as insufficient for assigning the value "College" — an error human readers would be unlikely to make. Along similar lines, inferring where a person grew up requires reading cues and competently weighing different pieces of information. This goes to say that the model sometimes approached the texts with a perhaps overly literal reading, struggling to integrate cues into broader inferences. In a rapidly evolving environment where new and more capable models are frequently released, it is unclear to what extent these particular observations will remain valid. Rather than giving definite predictions on which tasks models will do well on, our goal, then, is more modest: by exploring a wide array of tasks and variables, we hope to open readers' imagination as to the different kinds of information they *might* now be able to extract from text at scale. The true and only possible answer to the initial question is best summarized with Grimmer and Stewart's old mantra: "Validate, validate, validate." (2013, p. 271)

Another important insight from our results is that for some tasks, there may be much more economical tools than generative LLMs. In the case of *age*, for example, a simple regular expression performed almost as well as prompting our 70 billion parameter model. While it may be tempting to use prompting as a one-stop-shop for IE tasks, this might amount to a dramatic waste of resources when data sets are large. We recommend that researchers wanting to do IE carefully consider how the information they would like to extract is embedded in their texts. Sometimes, this embedding will be patterned in ways that lend themselves to other forms of extraction. In our case, it was our close reading of the text in the development phase that eventually led us to recognize how we could reliably extract the age of a deceased person. Similarly, we found that for one of our tasks (*gender*), fine-tuning a small language model led to similarly accurate results as prompting a large one (resonating findings by Ma et al. 2023; Chae and Davidson 2024). Therefore, even when prompting generative LLMs yields highly accurate results, this approach need not be the first choice.

Along similar lines, we found that our smaller model could handle some IE tasks just as or almost as well as the large one. Smaller models may therefore be another way to save both computational and logistical costs. Our analyses of twelve different IE tasks indicate that performance differences between models of different sizes don't follow a simple scaling function: some tasks that were "difficult" for the small model were "easy" for the large model. Others were similarly easy or difficult for both. The observed performance differences appear to depend on qualities of the tasks, such as whether the information is highly salient or more akin to a needle in a haystack. However, these observations remain qualitative and should not be interpreted as reliable predictions as to what tasks a small model can or can't handle. Instead, we recommend that researchers take an exploratory approach, beginning with smaller models. Our results suggest that there is a good chance that these might have similar accuracy or accuracy that could be sufficient in light of the analytic goals of the project.

When it comes to strategies for improving model output, the most significant choice was whether or not to use chain-of-thought prompting. The impact of this strategy, however, varied: it improved performance on a task where the model struggled to follow explicit instructions but led to declines for others. Asking for a specific output format or applying 1-shot prompting did not have much of an effect on our tasks. While beyond the scope of this paper, it is worth pointing out that there are a variety of other techniques and prompting strategies that researchers seeking to improve their results can explore. For instance, Brown et al. (2020) found positive effects of increasing the number of examples given to the model in few-shot prompting beyond one. Researchers might also want to consider logit biasing, a technique for adjusting the likelihood of specific tokens during text generation, effectively restricting or promoting certain model outputs. In a recent paper, Burnham (2024) found that this improved model performance on a social science classification task. Alternatively, or in addition, one could use thresholds for output tokens. That is, one might consider an information absent from the text unless the model's response tokens have a probability above a certain threshold. This might be an effective way to reduce false positives and to mitigate hallucinations (see, e.g., Plauth, Nguyen, and Trinh 2024). Prompt tuning (Li and Liang 2021; Lester, Al-Rfou, and Constant 2021), as discussed above, might be another way of improving model performance.

Finally, the overall high accuracy for extracting a variety of pieces of information should not distract from the fact that our results also uncovered a concerning weakness: models sometimes rely on semantic patterns learned from their training data to infer probable answers that have no justification in the text. Our most clear-cut example of this concerned *religion*, where even through explicit instruction, we could not reliably prevent models from making inferences based on identity markers like nationality, name, or origin. While this is a drastic case, it is easy to imagine how this tendency might affect other tasks, especially when it comes to extracting demographic and identity-related information. This not only raises ethical concerns (see, e.g., Joyce et al. 2021) but implies that the error in measures derived with a prompting approach will likely be non-random. This has major implications for downstream analyses. As Egami and colleagues (2024, 2023) recently demonstrated with regard to text classification, even if models produce labels that are highly accurate in the aggregate, non-random errors can lead to highly biased estimates in subsequent analyses. Therefore, we recommend that researchers carefully examine the kinds of errors that their models make and test whether errors are correlated with variables that are of theoretical interest. We also recommend exploring statistical tools for systematically addressing this challenging problem, at least as it relates to obtaining de-biased estimators in downstream analyses, such as the Design-based Supervised Learning (DSL) framework (see Egami et al. 2024, 2023).

## Discussion

Having extracted a variety of pieces of information with mixed but overall high accuracy, we see three core advantages of prompt-based IE over previous approaches relating to *accessibility*, *flexibility*, and *transparency and reproducibility*. However, based on our results, we also want to highlight a major risk of this approach, which concerns the likely *non-randomness of prediction errors*.

First, and perhaps most importantly, prompt-based IE is highly *accessible*. In a pre-GenAI era, doing IE required tedious human labor or would have required sociologists to familiarize themselves with a branch of highly specialized and often niche NLP methods. In retrospect, we think that there may have been real potential for sociological analysis in some of the specialized IE tools developed in NLP. To some extent, however, GenAI gives sociologists the chance to leapfrog over such tools in a manner that requires relatively few special skills. Most of the work we did for this study involved conceptually developing our variables, annotating, as well as editing prompts and examining results to steer the model toward producing desired outcomes. While it is worth familiarizing oneself with basic prompting strategies, prompt development is not a high art. As we mentioned above, we think that developing prompts from an existing codebook is relatively straightforward, and researchers can build on prompting patterns from other published work or prompt catalogs (see, e.g., White et al. 2023).

Furthermore, running inference with the models typically involves only a few lines of code. Perhaps the most difficult technical aspect of a project like ours is setting models up on an appropriate compute infrastructure. For few researchers, it will make sense to purchase the hardware necessary to run inference on even a moderately sized open-source generative LLM like the one we used. Therefore, running inference will typically require using an HPC cluster or other cloud compute environment. Increasing adoption of inference with generative LLMs will likely lead research institutions to provide guides and instructions for doing so, as some already have. Some may even provide graphical user interfaces that further reduce technical hurdles. That said, it is also important to point out that access to technical support and the hardware necessary for this kind of project will not be distributed equally. Researchers at some institutions will have clear advantages over others, especially when it comes to projects that require inference on a large number of documents. Nonetheless, we think that overall, GenAI makes running IE much more accessible for many sociologists, especially for those without specialized skills in NLP.

Leveraging GenAI for IE via prompts offers another key advantage over previous approaches, especially from the point of view of social science research: it is highly *flexible*. Rather than having to build on IE templates and ontologies developed by others, sociologists now have the opportunity to tailor coding schemes to their project and research question. Rather than adapting their work to a tool just because it is available, researchers will ideally be able to align models with their respective codebooks. As Davidson recently put it, GenAI can "facilitate sophisticated approaches to content analysis that avoid sacrificing the interpretative nuances that are easily lost when using conventional computational techniques" (2024, p. 5).

There is, of course, a limit here. As we illustrated in Figure 1 and as we noted in our discussion of sociologists' usage of event catalogs, IE templates are often more complex. Here, with the exception of listing survivors, we focused on extracting single pieces of information. Advanced IE tasks may involve retrieving multiple events from a text, encoding temporal or other meta information about these events, or identifying linked entities and tagging the relationship. It appears likely that such tasks would have proven more difficult for our model. However, it is also not certain that they would have provided an insurmountable challenge, for one can disaggregate most tasks into a series of sub-tasks, such as, say, identifying the sentence that relates two entities, and then running a series of prompts on that sentence to fill a template. We anticipate that future studies will more thoroughly explore these possibilities than we could here.

Third, we think that the use of prompts provides an opportunity for increasing *transparency* in the annotation of texts. This argument may appear surprising, as others have rightly pointed out that generative LLMs pose some challenges to replicability (Ollion et al. 2024; Spirling 2023). In particular, proprietary models can be modified or removed by third parties without notice, thereby thwarting any future replication efforts. We share these concerns, but we also think that GenAI offers some unique opportunities when it comes to being transparent about how a concept was measured. As discussed above, codebooks ideally provide complete and exhaustive instructions for anyone to replicate the codes used in a project. However, the reality is that most data annotation in social science projects occurs in groups, and such groups operate to some extent on implicitly shared understandings that may not be spelled out. Furthermore, there is effectively no way for readers or reviewers to assess the extent to which a codebook's instructions were correctly implemented on the training data unless they want to engage in data labeling themselves (which they rarely do). Meanwhile, data labeling and extracting information via prompts largely depend on two things, both of which ideally are publicly available: the prompt and the model.

The combination of an open-source model and what we call a promptbook goes a long way in giving anyone a clear sense of the data labeling process that underlies a study. Not only can readers easily examine the promptbook and assess its quality (which we invite the readers to do; see Appendix A), but this also opens up exciting new possibilities for prompt-sharing and building on other people's work. While previously, researchers could build on others' codebooks, this hasn't exactly been a common practice in sociology, and as indicated above, we have some doubts as to what extent most codebooks would suffice to truly replicate a concept. Meanwhile, promptbooks make it very easy to implement another researcher's measurement strategy for a certain concept simply by running their prompt against the model they chose. This might facilitate a new level of dialogue among researchers, where one project's conception of a complex concept (whether "populist rhetoric," "politeness," or "anger," to name a few examples) can easily be replicated in another. The point then isn't so much that GenAI is *necessary* for replication, but that it makes it *easier*, and therefore more *likely* to actually happen.

Fourth, despite these positive aspects, we also want to warn researchers that the likely *non-randomness of prediction errors* implies a major risk of this approach. Social scientists will typically extract information with generative LLMs to perform downstream analyses. When errors in the extracted information are correlated with theoretically relevant variables, they can introduce significant biases into downstream estimates. This risk remains present even when the extracted information is highly accurate overall (Egami et al. 2023, 2024). To be clear, this problem is not unique to variables generated with a prompting approach. Social scientists have long overlooked the fact that non-random errors in machine-generated predictions may bias downstream analysis (ibid.). However, the risks of doing so appear significantly greater with a new family of models that can draw on semantic patterns learned from exogenous data to infer responses. A model that infers religion based on a name despite being instructed not to do so effectively draws on its own prejudices. Researchers need to be aware that even if they have a largely accurate IE model, this accuracy may vary across subsections of their data, and, consequently, that extracted data may reflect the model's biases. This is not an easy problem to address and we commend recent efforts to give social scientists statistical tools for systematically tackling it (Egami et al. 2023, 2024).

Nonetheless, overall, our analyses demonstrate that generative LLMs can be turned into flexible and (for many tasks) useful IE engines. While it is good to be cautious in the context of a general hype around GenAI, we do think that this further opens the door for a revival of a sociological paradigm that treats texts as informants about events, people, relations, organizations, places, and other kinds of entities. Many of the highlights of this paradigm (e.g., Wang and Soule

2012, Tilly 1997, Mohr 1994, Mische & Pattison 2000) required large research collaborations and/or immense investments of labor and research money. With prompt-based IE, sociologists now have a comparatively affordable and scalable tool for collecting structured data from unstructured corpora. This could open new frontiers across a variety of research areas like studying methods used in scientific articles (Dagdelen et al. 2024), characteristics attributed to players in sports broadcasting (Foy and Ray 2019), definitions of relationships in interpersonal communication (McFarland & Wolff 2022), and many others.

## Conclusion

In theater, the term "promptbook" refers to the document that records all the information needed to produce a show. Details relating to script, acting, lighting, or sound are documented in order to leave as few uncertainties as possible as to how a performance should unfold. In the wake of the development of generative AI and the increasing adoption of prompt-based techniques in sociology, it seems fitting to us to borrow this term. Adapted to the social sciences, a promptbook should document the tasks involved in conducting an annotation or IE project.

Along with open-source models, the use and sharing of promptbooks could considerably increase the transparency of social scientific content analysis. It gives readers a direct chance to examine the authors' conceptual choices. In other words, it makes explicit what sometimes remains implicit in a researcher's work. Yet promptbooks have another important function: More informative than the raw annotations used to train models, and usually more explicit than the codebooks provided to research assistants, they allow us to *share* research operations involved in a process that has traditionally often remained hidden to others. Just as code-sharing allows researchers working with quantitative data to imitate, reflect, and improve on the solutions designed by their colleagues, promptbooks allow us to make the other dimensions of research more visible and exchangeable.

## References


Aher, Gati V., Rosa I. Arriaga, and Adam Tauman Kalai. 2023. "Using Large Language Models to Simulate Multiple Humans and Replicate Human Subject Studies." Pp. 337–71 in *Proceedings of the 40th International Conference on Machine Learning*. PMLR.

Alvero, A. J., Jinsook Lee, Alejandra Regla-Vargas, René Kizilcec, Thorsten Joachims, and anthony lising antonio. 2024. "Large Language Models, Social Demography, and Hegemony: Comparing Authorship in Human and Synthetic Text." *SocArxiv*. doi: 10.31235/osf.io/qfx4a.

Argyle, Lisa P., Ethan C. Busby, Nancy Fulda, Joshua R. Gubler, Christopher Rytting, and David Wingate. 2023. "Out of One, Many: Using Language Models to Simulate Human Samples." *Political Analysis* 31(3):337–51. doi: 10.1017/pan.2023.2.

Bail, Christopher A. 2024. "Can Generative AI Improve Social Science?" *Proceedings of the National Academy of Sciences* 121(21):e2314021121. doi: 10.1073/pnas.2314021121.

Baranick, Alana, Jim Sheeler, and Stephen Miller. 2005. *Life on the death beat: A handbook for obituary writers*. Marion Street Press.

Bearman, Peter, Robert Faris, and James Moody. 1999. "Blocking the Future: New Solutions for Old Problems in Historical Social Science." *Social Science History* 23(4):501–33. doi: 10.1017/S0145553200021854.

Bearman, Peter S., and Katherine Stovel. 2000. "Becoming a Nazi: A Model for Narrative Networks." *Poetics* 27(2):69–90. doi: 10.1016/S0304-422X(99)00022-4.

Beltagy, Iz, Matthew E. Peters, and Arman Cohan. 2020. "Longformer: The long-document transformer." *arXiv preprint arXiv:2004.05150*.

Bonikowski, Bart, Yuchen Luo, and Oscar Stuhler. 2022. "Politics as Usual? Antecedents of Radical-Right Frames in U.S. Electoral Discourse." *Sociological Methods & Research* 51(4):1721-1787.

Bourdieu, Pierre. 1988. *Homo Academicus*. Cambridge, UK: Polity Press.

Braun, Robert, and Ruud Koopmans. 2010. “The Diffusion of Ethnic Violence in Germany: The Role of Social Similarity.” *European Sociological Review* 26(1):111–23. doi: 10.1093/esr/jcp056.

Braun, Robert, and Ruud Koopmans. 2014. “Watch the Crowd: Bystander Responses, Trickle-Down Politics, and Xenophobic Mobilization.” *Comparative Political Studies* 47(4):631–58. doi: 10.1177/0010414013488544.

Broska, David, Michael Howes, Austin van Loon. 2024. The Mixed Subjects Design: Treating Large Language Models as Potentially Informative Observations. OSF preprint: https://doi.org/10.31235/osf.io/j3bnt_v3

Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. “Language Models Are Few-Shot Learners.” *arXiv:2005.14165*

Burnham, Michael. 2024. “Stance detection: a practical guide to classifying political beliefs in text.” *Political Science Research and Methods*. Published online. doi:10.1017/psrm.2024.35

Bytheway, Bill, and Julia Johnson. 1996. “Valuing Lives? Obituaries and the Life Course.” *Mortality* 1(2):219–34. doi: 10.1080/713685833.

Carley, Kathleen. 1993. “Coding Choices for Textual Analysis: A Comparison of Content Analysis and Map Analysis.” *Sociological Methodology* 23:75–126. doi: 10.2307/271007.

Carley, Kathleen. 1994. “Extracting Culture through Textual Analysis.” *Poetics* 22(4):291–312. doi: 10.1016/0304-422X(94)90011-6.

CDC. 2001. “HIV and AIDS — United States, 1981–2000.” *Morbidity and Mortality Weekly Report* 50(21): 429-456.

Chae, Youngjin and Thomas Davidson. 2024. "Large Language Models for Text Classification. From Zero-Shot Learning to Instruction-Tuning". Working paper.

Dagdelen, John, Alexander Dunn, Sanghoon Lee, Nicholas Walker, Andrew S. Rosen, Gerbrand Ceder, Kristin A. Persson and Anubhav Jain. 2024. “Structured information extraction from scientific text with large language models.” *Nature Communications* 15.1: 1418.

Davidson, Thomas. 2024. “Start Generating: Harnessing Generative Artificial Intelligence for Sociological Research.” *Socius: Sociological Research for a Dynamic World* 10:23780231241259651. doi: 10.1177/23780231241259651.

Do, Salomé, Étienne Ollion, and Rubing Shen. 2022. “The augmented social scientist: Using sequential transfer learning to annotate millions of texts with human-level accuracy.” *Sociological Methods & Research* 53(3): 1167-1200.

Dubey, Abhimanyu, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur et al. 2024. "The Llama 3 Herd of Models." *arXiv*. arXiv:2407.21783.

Dunivin, Zackary. 2024. “Scalable Qualitative Coding with LLMs: Chain-of-Thought Reasoning Matches Human Performance in Some Hermeneutic Tasks.” *arXiv.* arXiv:2401.15170.

Egami, Naoki, Musashi Hinck, Brandon M. Stewart, and Hanying Wei. 2024. “Using Large Language Model Annotations for the Social Sciences: A General Framework of Using Predicted Variables in Downstream Analyses.”. Working paper. https://naokiegami.com/paper/dsl_ss.pdf.

Egami, Naoki, Musashi Hinck, Brandon Stewart, and Hanying Wei. 2023. “Using Imperfect Surrogates for Downstream Inference: Design-based Supervised Learning for Social Science Applications of Large Language Models.” *Proceedings of the 37th International Conference on Neural Information Processing Systems*, Article 3000, 68589-68601. arXiv:2306.04746.

Fowler, Bridget, and Esperança Bielsa. 2007. “The Lives We Choose to Remember: A Quantitative Analysis of Newspaper Obituaries.” *The Sociological Review* 55(2):203–26. doi: 10.1111/j.1467-954X.2007.00702.x.

Foy, Steven L., and Rashawn Ray. 2019. “Skin in the game: Colorism and the subtle operation of stereotypes in men’s college basketball.” *American Journal of Sociology* 125.3: 730-785.

Franzosi, Roberto. 1989. “From words to numbers: A generalized and linguistics-based coding procedure for collecting textual data.” *Sociological Methodology* 19(1): 263-298.

Franzosi, Roberto. 1990. “Computer-assisted coding of textual data: An application to semantic grammars.” *Sociological Methods & Research* 19(2): 225-257.

Franzosi, Roberto. 1994. "From words to numbers: A set theory framework for the collection, organization, and analysis of narrative data." *Sociological Methodology* 24: 105-136.

Franzosi, Roberto. 1998. "Narrative as data: linguistic and statistical tools for the quantitative study of historical events." *International review of social history* 43:81-104.

Gao, Chen, Xiaochong Lan, Zhihong Lu, Jinzhu Mao, Jinghua Piao, Huandong Wang, Depeng Jin, and Yong Li. 2023. "S3: Social-Network Simulation System with Large Language Model-Empowered Agents." *arXiv*. arXiv:2307.14984.

Gerganov, Georgi. 2024. Llama.cpp: Inference of Meta's LLaMA model (and others) in pure C/C++. https://github.com/ggerganov/llama.cpp.

Gilardi, Fabrizio, Meysam Alizadeh, and Maël Kubli. 2023. "ChatGPT Outperforms Crowd Workers for Text-Annotation Tasks." *Proceedings of the National Academy of Sciences* 120(30):e2305016120. doi: 10.1073/pnas.2305016120.

Goldenstein, Jan, and Philipp Poschmann. 2019. "Analyzing Meaning in Big Data: Performing a Map Analysis Using Grammatical Parsing and Topic Modeling." *Sociological Methodology* 49(1):83–131. doi: 10.1177/0081175019852762.

Götz, Friedrich M., Rakoen Maertens, Sahil Loomba, and Sander van der Linden. 2023. "Let the Algorithm Speak: How to Use Neural Networks for Automatic Item Generation in Psychological Scale Development." *Psychological Methods*. doi: 10.1037/met0000540.

Gray, Robert and David L. Neuhoff. 1998. "Quantization." *IEEE Transactions on Information Theory*, 44(6): 2325–2383.

Grishman, Ralph. 2022. "Information Extraction." pp. 934-950 in: *The Oxford Handbook of Computational Linguistics,* edited by R. Mitkov. London: Oxford University Press.

Grishman, Ralph. 2019. "Twenty-five years of information extraction." *Natural Language Engineering* 25(6): 677-692. doi: 10.1017/S1351324919000512.

Grimmer, Justin, Margaret E. Roberts, and Brandon M. Stewart. 2022. *Text as Data: A New Framework for Machine Learning and the Social Sciences*. Princeton: Princeton University Press.

Grimmer, Justin, and Brandon M. Stewart. 2013. "Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts." *Political Analysis* 21(3): 267-297.

Hume, Janice. 2000. *Obituaries in American Culture*. Jackson: University Press of Mississippi.

Horton, John J. 2023. "Large Language Models as Simulated Economic Agents: What Can We Learn from Homo Silicus?" National Bureau of Economic Research. NEBR Working Papers. Working Paper 31122. doi: 10.3386/w31122.

Hwang, Jackelyn, and Nikhil Naik. 2023. "Systematic Social Observation at Scale: Using Crowdsourcing and Computer Vision to Measure Visible Neighborhood Conditions." *Sociological Methodology* 53(2):183–216. doi: 10.1177/00811750231160781.

Jacobs, Gilles, and Véronique Hoste. 2022. "SENTiVENT: Enabling Supervised Information Extraction of Company-Specific Events in Economic and Financial News." *Language Resources and Evaluation* 56(1):225–57. doi: 10.1007/s10579-021-09562-4.

Jallinoja, Riitta. 2011. "Obituaries as Family Assemblages." Pp. 78–91 in *Families and Kinship in Contemporary Europe: Rules and Practices of Relatedness*, edited by R. Jallinoja and E. D. Widmer. London: Palgrave Macmillan UK.

Jiang, Albert. Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot et al. 2024. "Mixtral of experts." *arXiv*. arXiv:2401.04088.

Joyce, Kelly, Laurel Smith-Doerr, Sharla Alegria, Susan Bell, Taylor Cruz, Steve G. Hoffman, Safiya Umoja Noble, and Benjamin Shestakofsky. 2021. "Toward a Sociology of Artificial Intelligence: A Call for Research on Inequalities and Structural Change." *Socius* 7:2378023121999581. doi: 10.1177/2378023121999581.

Jurafsky, Daniel, James H. Martin. 2024. "Information Extraction: Relations, Events, and Time." pp. 435-460 in *Speech and Language Processing*, edited by D. Jurafsky and J. H. Martin. https://web.stanford.edu/~jurafsky/slp3/20.pdf.

Karell, Daniel, Jeffrey Sachs, Ryan Barrett. 2023. "Synthetic Duality: A Framework for Analyzing Natural Language Generation's Representation of Social Reality." *SocArXiv*. https://osf.io/htxa4.

Kim, Junsol, and Byungkyu Lee. 2024. "AI-Augmented Surveys: Leveraging Large Language Models and Surveys for Opinion Prediction." *arXiv.* arXiv:2305.09620.

Knight, Carly. 2022. “When Corporations Are People: Agent Talk and the Development of Organizational Actorhood, 1890–1934.” *Sociological Methods & Research* 51(4):1634–80. doi: 10.1177/00491241221122528.

Krippendorff, Klaus. 2019. *Content Analysis: An Introduction to Its Methodology*. California: SAGE Publications, Inc.

Koleck, Theresa A., Caitlin Dreisbach, Philip E. Bourne, and Suzanne Bakken. 2019. “Natural Language Processing of Symptoms Documented in Free-Text Narratives of Electronic Health Records: A Systematic Review.” *Journal of the American Medical Informatics Association: JAMIA* 26(4):364–79. doi: 10.1093/jamia/ocy173.

Kozlowski, Austin and James A. Evans. 2024. “Simulating Subjects: The Promise and Peril of AI Stand-Ins for Social Agents and Interactions.” OSF preprint: https://doi.org/10.31235/osf.io/vp3j2

Kozlowski, Austin and Hyunku Kwon, James A. Evans. 2024. “In Silico Sociology: Forecasting COVID-19 Polarization with Large Language Models.” ArXiv preprint: arXiv:2407.11190

Law, Tina and Elizabeth Roberto. 2024. “Generative Multimodal Models for Social Science: an Application with Satellite and Streetscape Imagery.” SocArXiv preprint: https://doi.org/10.31235/osf.io/6jq32

Lester, Brian, Rami Al-Rfou, and Noah Constant. 2021. “The power of scale for parameter-efficient prompt tuning.” *arXiv preprint arXiv:2104.08691*.

Li, Xiang Lisa, and Percy Liang. 2021. “Prefix-tuning: Optimizing continuous prompts for generation.” *arXiv preprint arXiv:2101.00190*. Available at: https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-entities-guidelines-v5.6.6.pdf (accessed 11.30.2024)

Linguistic Data Consortium. 2024. "Annotation Tasks and Specifications." *Available at: https://www.ldc.upenn.edu/collaborations/past-projects/ace/annotation-tasks-and-specifications (accessed 11.30.2024)*

Long, Gary L. 1987. “Organizations and Identity: Obituaries 1856-1972.” *Social Forces* 65(4):964. doi: 10.2307/2579019.

Ma, Yubo, Yixin Cao, Yong Ching Hong, and Aixin Sun. 2023. “Large Language Model Is Not a Good Few-shot Information Extractor, but a Good Reranker for Hard Samples!” *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. 10572–10601

Makovi, Kinga, Ryan Hagen, and Peter Bearman. 2016. “The Course of Law: State Intervention in Southern Lynch Mob Violence 1882–1930.” *Sociological Science* 3:860–88. doi: 10.15195/v3.a37.

Martin, John Levi. 2000. “What Do Animals Do All Day?: The Division of Labor, Class Bodies, and Totemic Thinking in the Popular Imagination.” *Poetics* 27(2–3):195–231. doi: 10.1016/S0304-422X(99)00025-X.

McFarland, Daniel A., and Tom Wolff. 2022. "Writing into relationships." *Social Networks* 71: 96-114.

Mische, Ann, and Philippa Pattison. 2000. “Composing a Civic Arena: Publics, Projects, and Social Settings.” *Poetics* 27(2):163–94. doi: 10.1016/S0304-422X(99)00024-8.

Mohr, John W. 1994. “Soldiers, Mothers, Tramps and Others: Discourse Roles in the 1907 New York City Charity Directory.” *Poetics* 22(4):327–57. doi: 10.1016/0304-422X(94)90013-2.

Mohr, John W., and Vincent Duquenne. 1997. “The Duality of Culture and Practice: Poverty Relief in New York City, 1888-1917.” *Theory and Society* 26(2/3):305–56.

Mohr, John W., and Helene K. Lee. 2000. “From Affirmative Action to Outreach: Discourse Shifts at the University of California.” *Poetics* 28(1):47–71. doi: 10.1016/S0304-422X(00)00024-3.

Mohr, John W., Robin Wagner-Pacifici, Ronald L. Breiger, and Petko Bogdanov. 2013. “Graphing the Grammar of Motives in National Security Strategies: Cultural Interpretation, Automated Text Analysis and the Drama of Global Politics.” *Poetics* 41(6):670–700. doi: 10.1016/j.poetic.2013.08.003.

Naik, Nikhil, Scott Duke Kominers, Ramesh Raskar, Edward L. Glaeser, and César A. Hidalgo. 2017. “Computer Vision Uncovers Predictors of Physical Urban Change.” *Proceedings of the National Academy of Sciences* 114(29):7571–76. doi: 10.1073/pnas.1619003114.

New York Times. 2022. “How does the New York Times decide who gets an obituary?”. https://www.nytimes.com/article/new-york-times-obituary-process.html.

Ollion, Étienne, Rubing Shen, Ana Macanovic, and Arnault Chatelain. 2024. “The Dangers of Using Proprietary LLMs for Research.” *Nature Machine Intelligence* 6(1):4–5. doi: 10.1038/s42256-023-00783-6.

Palmer, Alexis, Noah A. Smith, and Arthur Spirling. 2024. “Using Proprietary Language Models in Academic Research Requires Explicit Justification.” *Nature Computational Science* 4(1):2–3. doi: 10.1038/s43588-023-00585-1.

Park, Joon Sung, Joseph O’Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. “Generative Agents: Interactive Simulacra of Human Behavior.” Pp. 1–22 in *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, *UIST ’23*. New York, NY, USA: Association for Computing Machinery.

Plauth, Benjamin, Khanh Xuan Nguyen, and Tu Trinh. 2024. “Probabilities of Chat LLMs Are Miscalibrated but Still Predict Correctness on Multiple-Choice Q&A.” *arXiv:2402.13213v2.*

Renze, Matthew and Erhan Guven 2024. “The effect of sampling temperature on problem solving in large language models.” *arXiv preprint arXiv:2402.05201*.

Smith, Tammy. 2007. “Narrative Boundaries and the Dynamics of Ethnic Conflict and Conciliation.” *Poetics* 35(1):22–46. doi: 10.1016/j.poetic.2006.11.001.

Spirling, Arthur. 2023. “Why Open-Source Generative AI Models Are an Ethical Way Forward for Science.” *Nature* 616(7957):413. doi: 10.1038/d41586-023-01295-4.

Stuhler, Oscar. 2021. “What’s in a Category? A New Approach to Discourse Role Analysis.” *Poetics* 88:101568. doi: 10.1016/j.poetic.2021.101568.

Stuhler, Oscar. 2024. “The Gender Agency Gap in Fiction Writing (1850 to 2010).” *Proceedings of the National Academy of Sciences* 121(29):e2319514121. doi: 10.1073/pnas.2319514121.

Tilly, Charles. 1995. *Popular Contention in Great Britain, 1758-1834*. Cambridge, MA: Harvard University Press.

Tilly, Charles. 1997. “Parliamentarization of Popular Contention in Great Britain, 1758-1834.” *Theory and Society* 26(2/3):245–73.

Tilly, Charles. 2002. “Event Catalogs as Theories.” *Sociological Theory* 20(2):248–54. doi: 10.1111/1467-9558.00161.

Törnberg, Petter. 2023. “ChatGPT-4 Outperforms Experts and Crowd Workers in Annotating Political Twitter Messages with Zero-Shot Learning.” arXiv. arXiv:2304.06588.

Underwood, Ted. 2003. “Using GPT-4 to measure the passage of time in fiction,” *Using Large Digital Libraries to Advance Literary History* (blog). March 19. https://tedunderwood.com/2023/03/19/using-gpt-4-to-measure-the-passage-of-time-in-fiction/.

Wang, Dan J., and Sarah A. Soule. 2012. “Social Movement Organizational Collaboration: Networks of Learning and the Diffusion of Protest Tactics, 1960-1995.” *American Journal of Sociology* 117(6).

Wang, Dan J., and Sarah A. Soule. 2016. “Tactical innovation in social movements: The effects of peripheral and multi-issue protest.” *American Sociological Review* 81(3): 517-548.

Wang, Yanshan, Liwei Wang, Majid Rastegar-Mojarad, Sungrim Moon, Feichen Shen, Naveed Afzal, Sijia Liu, Yuqun Zeng, Saeed Mehrabi, Sunghwan Sohn, and Hongfang Liu. 2018. “Clinical Information Extraction Applications: A Literature Review.” *Journal of Biomedical Informatics* 77:34–49. doi: 10.1016/j.jbi.2017.11.011.

Wei, Jason, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. "Chain-of-thought prompting elicits reasoning in large language models." *Advances in neural information processing systems* 35:24824-24837. arXiv:2201.11903.

White, Jules, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C. Schmidt. 2023. "A prompt pattern catalog to enhance prompt engineering with ChatGPT." *arXiv*. arXiv:2302.11382.

Zhu, Xunyu, Jian Li, Yong Liu, Can Ma, and Weiping Wang. 2024. “A Survey on Model Compression for Large Language Models.” *arXiv:2308.07633*.

# Appendix A. Promptbook

Below, we print our promptbook. It contains all prompts that we used to extract the information described for our 12 variables. We also provide the system prompt that we used for all but one of the scenarios. The promptbook here reports the prompts used in a 0-shot scenario where we use chain-of-thought prompting and ask for the response to be formatted in plain text. All prompts will be made available in a replication repository (see next section for Model details).

| **Variable** | **Prompt** |
|---|---|
| System prompt | `You are a highly efficient information detection and extraction engine, specialized in analyzing natural language data.`<br><br>`You value accuracy: when the user asks you to extract certain information from given text data, you will try your best to adhere to what is directly mentioned in the text and the extraction criteria.`<br><br>`You value efficiency: your responses will be very concise, because they will be stored as values in a dataset. These responses will also strictly follow formatting conventions specified in the extraction prompt.` |
| Age in years | `Below I will provide an obituary of a deceased person.`<br><br>`Based on both the title, publication date, and the obituary text, infer the person’s age at death in years. You should respond with a numeric value. If the person's age is not inferable, respond with "9999".`<br><br>`-`<br><br>`You should include one or multiple quotes from the text as evidence for your response, as well as a rationale. Please clearly mark the evidence, rationale and response in the following order: evidence, rationale, response.`<br><br>`Please format your response in plain text and place each item inside quotation marks, like this:`<br><br>`EVIDENCE: "";`<br><br>`RATIONALE: "";`<br><br>`RESPONSE: "".`<br><br>`Here is the obituary date, title, and text: {print obituary}` |

| | |
|---|---|
| Military service | Below I will provide an obituary of a deceased person.<br><br>Does the obituary mention that the person served in the military? If so, respond with "yes". Otherwise, respond with "not mentioned". If a person served in a foreign (that is non-US) military, also respond with "yes". Please limit your response to only one of these two codes: "yes", "not mentioned".<br><br>Advisory roles do not count as having served in the military.<br><br>–<br><br>You should include one or multiple quotes from the text as evidence for your response, as well as a rationale. Please clearly mark the evidence, rationale and response in the following order: evidence, rationale, response.<br><br>Please format your response in plain text and place each item inside quotation marks, like this:<br><br>EVIDENCE: "";<br><br>RATIONALE: "";<br><br>RESPONSE: "".<br><br>Here is the obituary date, title, and text: {print obituary} |
| Cause of death | Below I will provide an obituary of a deceased person.<br><br>If the text details the deceased person's cause of death (e.g.: “died after a long battle with cancer”), record the cause of death exactly as it is stated in the text (e.g.: "cancer").<br><br>If multiple causes of death are mentioned, record them all, and separate each cause with a comma (e.g.: "heart disease, stroke".)<br><br>If the cause of death is unknown or not mentioned, respond with "not mentioned".<br><br>–<br><br>You should include one or multiple quotes from the text as evidence for your response, as well as a rationale. Please clearly mark the evidence, rationale and response in the following order: evidence, rationale, response.<br><br>Please format your response in plain text and place each item inside quotation marks, like this:<br><br>EVIDENCE: "";<br><br>RATIONALE: "";<br><br>RESPONSE: "". |

| | |
|---|---|
| | Here is the obituary date, title, and text: {print obituary} |
| Number of children | Below I will provide an obituary of a deceased person.<br><br>Based on the text, infer the number of children that the deceased person had. This number should include all children, whether biological, adopted or stepchildren.<br><br>If no children are mentioned, or if the obituary states that the person had no children, respond with "0".<br><br>If facts listed in the obituary logically imply that the deceased person had a child or multiple children, but it is not inferable how many, respond with ">1". For example, an obituary that mentions a person's grandchildren but does not list their children should be coded as ">1".<br><br>–<br><br>You should include one or multiple quotes from the text as evidence for your response, as well as a rationale. Please clearly mark the evidence, rationale and response in the following order: evidence, rationale, response.<br><br>Please format your response in plain text and place each item inside quotation marks, like this:<br><br>EVIDENCE: "";<br><br>RATIONALE: "";<br><br>RESPONSE: "".<br><br>Here is the obituary date, title, and text: {print obituary} |
| Highest level of education | Below I will provide an obituary of a deceased person.<br><br>Based on what the text says about this person's education, record the highest level of education of the deceased person as only one of the following codes: “Less than high school”, “High school”, “Some college”, “College”, “Masters, PhD, or equivalent”, “not inferable”<br><br>When giving your response, consider the following rules:<br><br>1) Generally, do not infer this person’s level of education from their occupation without explicit statements in the text about the degrees they obtained. The only exceptions to this are people who teach at universities, people who practice law, or those who practice medicine. In those cases, respond with ”Masters, PhD, or equivalent”.<br><br>2) If the text states that the person “attended” or “went to" (or similar phrases) a college or university but does not specify what degrees they obtained, or whether or not they graduated, this is sufficient to apply the code ”College”. Similarly, if the text |

| | |
|---|---|
| | mentions a high school as the highest place of education but does not mention whether the person graduated, respond with ”High School”.<br><br>3) If, in contrast, the text states that the person dropped out of or did not graduate from, or otherwise failed to obtain their degree from some university or college, use the code ”Some college”. The same applies to people who did not finish high school, in which case you put ”Less than high school”.<br><br>3) For the code ”Masters, PhD, or equivalent”, equivalents include professional or graduate–level degrees obtained at post–baccalaureate education institutions, such as law schools, medical schools, theological seminaries and business schools. Attendance at these post–baccalaureate institutions are considered as ”Masters, PhD, or equivalent”.<br><br>4) “Taking classes” at a college is not synonymous to “attending” a college. If the text says that a person only “took classes” at a college or university, this is coded as ”Some College”. Meanwhile, if the text mentions that the person “went to” a college, this would be coded as ”College”.<br><br>If the education level is not inferable, simply respond with "not inferable".<br><br>–<br><br>You should include one or multiple quotes from the text as evidence for your response, as well as a rationale. Please clearly mark the evidence, rationale and response in the following order: evidence, rationale, response.<br><br>Please format your response in plain text and place each item inside quotation marks, like this:<br><br>EVIDENCE: "";<br><br>RATIONALE: "";<br><br>RESPONSE: "".<br><br>Here is the obituary date, title, and text: {print obituary} |
| Institutions of higher education attended | Below I will provide an obituary of a deceased person.<br><br>Record all institutions of higher education that the person obtained a degree from (i.e., universities, colleges, or graduate & professional schools), exactly as written in the text. If the text indicates that this person attended some institution as a student, but did not complete their degree, record this institution as well. When giving your response, consider the following rules:<br><br>1) Do not include high schools or college preparatory schools. |

| | |
|---|---|
| | 2) Do not include institutions that the person's friends, family, coworkers or partners attended, unless the deceased person also attended them.<br><br>3) Obituaries may describe decedents who were employed at academic institutions, such as instructors, scientists, university administrators and coaches. You must distinguish higher education institutions that this person studied at from those that this person worked at. Only institutions where the person studied should be considered in your response. Do not record higher education institutions only because the person worked, taught, or held a job there. For example, if the text says "after transferring from University 1 to study mathematics at University 2, he eventually got a master's degree from University 3. He became a head coach at University 4 and taught sports science at University 5", your response should only include Universities 1, 2 and 3, but not University 4.<br><br>If the text does not mention any institutions of higher education that the person attended, simply respond with "none".<br><br>If your response is a list of two or more institutions, please separate each institution with a comma (e.g.: "university 1, university 2, university 3").<br><br>–<br><br>You should include one or multiple quotes from the text as evidence for your response, as well as a rationale. Please clearly mark the evidence, rationale and response in the following order: evidence, rationale, response.<br><br>Please format your response in plain text and place each item inside quotation marks, like this:<br><br>EVIDENCE: "";<br><br>RATIONALE: "";<br><br>RESPONSE: "".<br><br>Here is the obituary date, title, and text: {print obituary} |
| Survivors | Below I will provide an obituary of a deceased person.<br><br>Extract the family roles of all the people who the obituary lists as the survivors of the deceased person.<br><br>For example, if the obituary states "Besides his daughter Rachel, he is survived by his wife Olivia, another daughter, Jane Doe of Brooklyn, NY, a son named John, his two brothers Jack and Peter, a niece, and three grandchildren", you should list the family roles as: "1 wife, 2 daughters, 1 son, 2 brothers, 1 niece, 3 grandchildren". |

| | |
|---|---|
| | The text may mention family relations outside of its discussion of survivors, such as former spouses who had been divorced from the deceased person, or family members whose living status is unspecified. Do not include these family roles in your response unless they are also listed as a survivor of the deceased person.<br><br>If the text does not discuss someone's survivors, or if the text states that the deceased person had no survivors, simply respond with "none".<br><br>–<br><br>You should include one or multiple quotes from the text as evidence for your response, as well as a rationale. Please clearly mark the evidence, rationale and response in the following order: evidence, rationale, response.<br><br>Please format your response in plain text and place each item inside quotation marks, like this:<br><br>EVIDENCE: "";<br><br>RATIONALE: "";<br><br>RESPONSE: "".<br><br>Here is the obituary date, title, and text: {print obituary} |
| Gender | Below I will provide an obituary of a deceased person.<br><br>Based on the text, infer the gender of the deceased person. Provide a one-word response from only one of the following options: "male", "female", "other".<br><br>–<br><br>You should include one or multiple quotes from the text as evidence for your response, as well as a rationale. Please clearly mark the evidence, rationale and response in the following order: evidence, rationale, response.<br><br>Please format your response in plain text and place each item inside quotation marks, like this:<br><br>EVIDENCE: "";<br><br>RATIONALE: "";<br><br>RESPONSE: "".<br><br>Here is the obituary date, title, and text: {print obituary} |
| Occupation phrase | Below I will provide an obituary of a deceased person.<br><br>Based on the title and text, respond with either one or, at maximum, two phrases that best summarize the occupation of the |

deceased person. Occupational phrases can be single words ("biologist") or multiple words (e.g.: "business executive", "software engineer", "social worker"). When you respond, consider the following six rules:

1) Occupational phrases will often appear directly in the text. If the text does not use such phrases, but instead mentions specific job titles (e.g. “executive vice president of the commercial division and founder of XYZ Corporation”), respond with a phrase that summarizes these job titles (e.g.: "business executive").

2) Note that at times, obituaries will mention lower level positions that the deceased person began their career at. However, we only want to code the occupations or professional positions most heavily emphasized by the obituary.

3) Positions in voluntary associations, committees and social clubs do not qualify as an occupation.

4) Multiple phrases should only be used if the person engaged in substantively distinct professional positions (e.g.: "actor, filmmaker"). Multiple phrases should not be used if the professional positions are relatively similar (e.g.: "business executive, manager"). If obituaries mention more than two notable professional positions, only those two should be recorded that the obituary puts the most emphasis on.

5) For athletes, we record the sport (if specified). For scholars, we record the field (if specified).

6) If an obituary does not mention an occupation or anything like that, put "not mentioned".

If your response is a list of two occupational phrases, please separate each phrase with a comma (e.g.: "occupational phrase 1, occupational phrase 2").

–

You should include one or multiple quotes from the text as evidence for your response, as well as a rationale. Please clearly mark the evidence, rationale and response in the following order: evidence, rationale, response.

Please format your response in plain text and place each item inside quotation marks, like this:

EVIDENCE: "";

RATIONALE: "";

RESPONSE: "".

Here is the obituary date, title, and text: {print obituary}

| | |
|---|---|
| Origin | Below I will provide an obituary of a deceased person.<br><br>Record the municipality that the person was born in or grew up in. If multiple municipalities are mentioned, only insert that which appears to be the most significant according to the obituary. For instance, if the obituary indicates that a person was born somewhere, but moved and grew up somewhere else, record the place where they grew up.<br><br>For U.S. locations, infer and include the municipality and the state in your response if these are not directly mentioned in the text. For instance, a mention of “Crown Heights” in the text should be recorded as “Brooklyn, NY”. For locations outside of the U.S., record the province/state and country in the same format (e.g.: “Munich, Germany”). If only higher level locations are mentioned like countries, record these as they appear in the text (e.g.: “Florida” or “Switzerland”). Generally, your answer should be formatted as "Municipality, US-State" or "Municipality, non-US country".<br><br>If a place or municipality does not exist anymore (e.g., “Czechoslovakia”, still record it as it appears in the text.<br><br>It is important that the text mentions that the deceased person grew up in, spent part of their youth, or was born in the respective place. If no such place is mentioned, respond with “not mentioned”. Do not record other places only because they are mentioned in the text.<br><br>-<br><br>You should include one or multiple quotes from the text as evidence for your response, as well as a rationale. Please clearly mark the evidence, rationale and response in the following order: evidence, rationale, response.<br><br>Please format your response in plain text and place each item inside quotation marks, like this:<br><br>EVIDENCE: "";<br><br>RATIONALE: "";<br><br>RESPONSE: "".<br><br>Here is the obituary date, title, and text: {print obituary} |
| Place lived last | Below I will provide an obituary of a deceased person.<br><br>Record the municipality that the person lived last as described in the obituary.<br><br>If the obituary mentions that the person died in one place but had been living in another place, record the municipality they had been living in.<br><br>If the obituary mentions that the person died somewhere but makes |

| | |
|---|---|
| | no further clarification on whether they had been living somewhere else, assume that the municipality they died in is the municipality they had been living in.<br><br>For U.S. locations, infer and include the municipality and the state in your response if these are not directly mentioned in the text. For instance, a mention of “Crown Heights” in the text should be recorded as "Brooklyn, NY". For locations outside of the U.S., record the province/state and country in the same format (e.g.: "Munich, Germany"). If only higher level locations are mentioned like countries, record these as they appear in the text (e.g.: "Florida", "Switzerland"). Generally, your answer should be formatted as "Municipality, US-State" or "Municipality, non-US country".<br><br>If no place is mentioned, respond with "not mentioned".<br><br>–<br><br>You should include one or multiple quotes from the text as evidence for your response, as well as a rationale. Please clearly mark the evidence, rationale and response in the following order: evidence, rationale, response.<br><br>Please format your response in plain text and place each item inside quotation marks, like this:<br><br>EVIDENCE: "";<br><br>RATIONALE: "";<br><br>RESPONSE: "".<br><br>Here is the obituary date, title, and text: {print obituary} |
| Religion | Below I will provide an obituary of a deceased person.<br><br>If mentioned, respond with the deceased person’s religious affiliation. In your response, consider the following three rules:<br><br>1) A person’s religious affiliations must be mentioned explicitly. Only a person’s voluntary association with religious organizations or lifestyles (e.g., giving to a Catholic charity, careers or active roles at places of worship or religious associations) is sufficient to code religion. What is not sufficient to code religion includes: parents' religious affiliations, nationalities, ethnic identity markers, and employment at religiously-affiliated public service providers (e.g. working at a Jewish hospital, or at Catholic Charities).<br><br>2) If the text names a religious denomination, record it exactly as phrased. For instance, if the text mentions “roman-catholic” you should respond with "roman-catholic". However, if it simply mentions “catholic”, respond with "catholic". If multiple religions are mentioned, record these and separate them by a comma |

(e.g.: "buddhist, catholic").

3) If the text makes explicit references to the person's voluntary affiliation with some religion, but does not name their specific denomination, provide a more general term. A mention of a person going to "mosque," for instance, is not sufficient to code for, say, "sunni," but is sufficient to code as "muslim."

If no religious affiliation is mentioned, put "not mentioned".

–

You should include one or multiple quotes from the text as evidence for your response, as well as a rationale. Please clearly mark the evidence, rationale and response in the following order: evidence, rationale, response.

Please format your response in plain text and place each item inside quotation marks, like this:

EVIDENCE: "";

RATIONALE: "";

RESPONSE: "".

Here is the obituary date, title, and text: {print obituary}

## Appendix B. Model details

Inference was run using the Llama.cpp library (Gerganov 2024) via the llama-cpp-python Python wrapper (Version 0.2.83). Our two models are 5-bit quantized versions of Llama 3 Instruct 70B and 8B made available on the Model Hub at huggingface.co at "Bartowski/Meta-Llama-3-70B-Instruct-Q5_K_M.gguf" and "Bartowski/Meta-Llama-3-8B-Instruct-Q5_K_M.gguf." The former requires about 50 GB of disk space while the latter requires about 6 GB. Models were run using two NVIDIA V100 PCIe GPUs with 32 GB VRAM each. We used a temperature of 0 to have the model output the most likely tokens, leaving other hyperparameters at their default values.

# Appendix C. Baseline approaches

In this section, we give details on each of our baseline approaches.

As discussed in the main text, for *age*, we simply extracted the first two- or three-digit number that appeared in the obituary. While we had initially tried more complex approaches, this turned out to be highly accurate, as the decedent's age was often signaled either in the obituary headline or in the first few sentences where their death is announced.

For *cause of death*, we compiled a dictionary of phrases that we had found to frequently precede the stated cause of death in our development set. We then used regular expressions to extract all character strings that appear after such phrases and before the end of a sentence. For example, if a text has a sentence such as "his death was attributed to complications from heart surgery.", we would extract the string "complications from heart surgery." We iteratively adapted the search strings after testing them on our development set. The measures we report in the main paper are based on the results for the test set. The following is the full list of our regular expression strings, some of which account for some flexibility:

- "died\\s+(\\w+\\s+){0,3}of"[1]
- "died\\s+(\\w+\\s+){0,3}from"
- "died\\s+(\\w+\\s+){0,4}after"
- "died\\s+(\\w+\\s+){0,4}in\\s+[a-zA-Z]+"
- "died by"
- "was killed"
- "(his|her) death followed"
- "died\\s+(\\w+\\s+){0,3}following"
- "the cause was"
- "the cause of death was"
- "caused by"

[1] "\\s+(\\w+\\s+){0,3}" is a regex pattern that stands for "0, 1, 2 or 3 words". A search string like "died\\s+(\\w\\s+){0.3}of" would include phrases like "died of", "died suddenly of", "died on Thursday of", and "died in Cedar Rapids, of".

- "death was attributed to"
- "suffered a"
- "suffered from"
- "suffering from"

We decided not to “clean” the extracted results by using a dictionary of possible causes of death or using other regex patterns. We instead manually compared regex results to our ground-truth annotations to produce accuracy scores.

For three categorical variables with a fixed set of possible responses (gender, military service, and number of children), we fine-tuned Longformer models (Beltagy, Peters, and Cohan 2020) for text classification. The Longformer is a transformer model based on RoBERTa (Liu et al 2019) that is designed to process longer texts. We labeled all cases in our 200-case test set by applying 10-fold cross validation. That is, we split these cases into 10 different sets, then fined-tuned the model on the cases in 9 of these, producing prediction labels for the respectively held-out set. We then computed accuracy and F1 scores based on the 200 test set the same way we did in our prompt-based approaches.

Using the same k-fold approach, we did ad-hoc hyperparameter searches for each variable, varying batch size, number of epochs, and learning rates, optimizing on the F1 score. Following Longformer’s original documentation (Beltagy, Peters, and Cohan 2020), we used the Adam optimizer with linear warmup and decay, with steps being equal to 10% of the training steps. We set the maximum possible sequence length to 4096 tokens to accommodate the longest obituary texts.

# Appendix D. One-shot prompting

This section provides more details about how we implemented one-shot prompting. In using GenAI for question-answering tasks, the concept of one-shot prompting involves giving the model one example input and one example output to demonstrate the task, similar to how new tasks are communicated to humans (Brown et al. 2020). In our case, the instruction part of the prompts we provide the model with is exactly the same as in the corresponding zero-shot scenario. However, rather than appending the obituary we want to analyze to the prompt, we append an example obituary to the prompt and then simulate a correct model response. Simulated responses follow the expected formatting requirements (JSON and chain-of-thought).

The example obituaries were hand-picked from the development set. That is, we chose example obituaries and wrote example answers based on the specific requirements of each task. For the example inputs, we selected obituaries from the development set that fit two criteria. First, we chose ones that were relatively short. Second, we chose ones that had a very unambiguously correct answer for the task at hand. We ended up focusing on four example obituaries, which, collectively, were sufficient to fulfill these criteria for each of our twelve tasks.

Our one-shot implements a chain-of-thought scenario. Therefore, we selected the relevant evidence and wrote up a rationale for our simulated model response. We combined these features with the answer and assembled them in a JSON. Generally, the rationale we used was along the lines of "the obituary directly states X, therefore the answer is X." For inference & summarization tasks where the example obituaries require a classificatory decision to be made to exclude plausible but incorrect items (e.g. early-career jobs in occupation, education institutions taught at but not studied at, education levels that may be implied from a person's profession but are not confirmed by the text, geographic locations mentioned that are not the origin or place lived last), we qualify our rationale with additional logics, such as "Although Y is mentioned, it does not count as a case of X because…", or "The obituary does not mention the thing that would fit [counterfactual to category X], therefore category X is chosen."

## Appendix E. Model verbosity and post-processing

In our analyses, we noticed that the model would sometimes add an introductory line before the response (e.g. “Here is the response …”) or additional notes after a response. This did not pose a challenge for extracting the answers, but we decided to examine this behavior systematically by coding model for one of our scenarios (JSON, chain-of-thought, 0-shot). Table A1 shows the relative frequency of verbosity for each variable, together with the average toke length of the response. Results indicate that this behavior is extremely rare for the 70B model. Only 2 out of 2,400 responses were affected. For the smaller 8B model, verbosity is slightly more common with a rate of 0 to 4% of the cases being affected, depending on the task (20 of 2400 responses in total).

We note that model’s verbosity did not create any obstacles for extracting the relevant piece of information for validation. To evaluate the 22,600 model responses (9 models, 12 tasks, 200 cases, plus 1,000 cases from baseline approaches), we proceeded as follows: First, we matched all responses that were exactly the same as in our gold standard test set. To eliminate issues concerning punctuation signs or minor typographic variations like trailing commas or quotes, we used regular expressions and, for character variables, matched cases that differed by no more than a few characters via Levenshtein distance. When the variable was formatted as a list (e.g., Mr. Doe is survived by "1 wife, 1 son, 1 brother"), we looked for all possible combinations that could match the information in the gold standard. With these three steps, we matched around three fourths of the model responses to the human-annotated responses. For the remaining roughly 5,000 responses that did not match, we resorted to manual validation of whether the response was still correct, incorrect, or, for some variables, partially correct. Most of these cases were incorrect responses or stemmed from variables like *occupation* that can have semantically correct of lexically distinct responses. For instance, if we recorded that someone was a “politician,” but the model recorded them as “lawmaker,” we would still consider the response to be correct. To illustrate this further, other examples we would treat as correct include cases like: “Author” and "author, writer"; “diplomat” and "ambassador”; “news reporter, critic” and “reporter, television critic”; “actress, cabaret performer” and “actress, cabaret singer” and so forth.

| **Model scenario** | | | |
|---|---|---|---|
| Size | | 8B | 70B |
| Requested JSON format | | Yes | Yes |
| Chain-of-thought applied | | Yes | Yes |
| Shots | | 0 | 0 |
| **Task** | **Measure** | | |
| Age in years (integer) | Verbosity | 0% | 0% |
| | Response length (median) | 28 | 32.5 |
| Gender (categorical) | Verbosity | 0% | 0% |
| | Response length (median) | 57 | 55.5 |
| Cause of death (text string) | Verbosity | 1% | 0% |

| | | | |
|---|---|---|---|
| | Response length (median) | 40 | 41 |
| Origin (pre-formatted text string) | Verbosity | .5% | .5% |
| | Response length (median) | 46.5 | 50 |
| Place lived last(pre-formatted text string) | Verbosity | 0% | 0% |
| | Response length (median) | 51.5 | 48 |
| Military (binary) | Verbosity | .5% | 0% |
| | Response length (median) | 58.5 | 32 |
| Children (integer) | Verbosity | 0% | 0% |
| | Response length (median) | 48 | 44 |
| Survivors (pre-formatted text string) | Verbosity | 1% | 0% |
| | Response length (median) | 55 | 59.5 |
| Occupation (text string) | Verbosity | 1% | 0% |
| | Response length (median) | 88.5 | 79 |
| Institutions of higher education attended (text string) | Verbosity | 1.5% | .5% |
| | Response length (median) | 60 | 54 |
| Highest level of education (categorical) | Verbosity | .5% | 0% |
| | Response length (median) | 59 | 59 |
| Religious affiliation (text string) | Verbosity | 4% | 0% |
| | Response length (median) | 57.5 | 36.5 |

**Table E1. Prevalence of Verbosity and Token Length**

*Note: This table shows the prevalence of verbosity in percent and the median token length for our 200-case test set as predicted by both models in one scenario (JSON, chain-of-thought, 0-shot). Verbosity is defined as the model adding introductory statements or endnotes outside of the bounds of the formatted response.*

# References

Beltagy, Iz, Matthew E. Peters, and Arman Cohan. 2020. "Longformer: The long-document transformer." *arXiv preprint arXiv:2004.05150*.

Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. "Language Models Are Few-Shot Learners." *arXiv:2005.14165*

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Man- dar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized bert pretraining ap- proach. *arXiv preprint*, abs/1907.11692.