On the Dangers of Naïve Replication: The Case of Implicature

Anil Korde
University of Maryland
akorde@umd.edu

Philip Resnik
University of Maryland
resnik@umd.edu

Abstract

Other people's code, data, and definition of a language task often provide the groundwork for new research efforts. The work we present here began as a straightforward investigation of conversational implicature, a central aspect of natural dialogue, starting with updating a prior method to employ more recent LLMs. But differences in results with the work we were replicating led to a deep dive into why those differences were occurring, and this led us to consider more carefully what it means to begin working on a topic with prior work "as a starting point". We describe our process, what we found, and lessons suggested about data quality, task definition, and the current pace of change in NLP.

1 Introduction

Conversational implicature (Grice, 1975) is a ubiquitous phenomenon in conversation, and as such it is highly relevant for conversational AI using large language models. Just as for other language-related capabilities, today's standard paradigm for progress is to use a well defined computational task, together with a benchmark dataset and evaluation metrics, to establish the current state of the art and then adapt or introduce new methods to improve it.

The standard approach is not without its problems, however. Tasks or metrics sometimes turn out to have problems with *measurement validity*, i.e. whether a measurement is actually measuring what we want measured—this has arisen, for example, in natural language inference (Poliak et al., 2018) and topic modeling (Hoyle et al., 2021). Datasets can produce results that don't generalize well. Data contamination may inflate estimates of system performance.

This paper began as an investigation of conversational implicature, aimed at building on prior methods and benchmarking introduced by Ruis et al. (2024). In the end, however, what emerged is a case study contributing to the literature on the pitfalls of uncritically accepting the prompts and data from prior work as a starting point. In the sections that follow, we begin by providing relevant background on the topic of conversational implicature and discuss our attempt to replicate Ruis et al. (2024). We then shift, based on what we found, to a meta-level discussion that leads us to highlight the more general lessons we think this effort turned out to offer about data quality, task definition, and ultimately, we would argue, the pace of change in NLP.

2 Background

The idea of conversational implicature was introduced by Grice (1975). He presents the idea of the Cooperative Principle: that utterances in a conversation are driven by the shared goal of moving the conversation forward. He also states a number of maxims by which the Cooperative Principle is realized. Deliberately violating these maxims, he then argues, is how conversational implicature arises. For instance, in the following exchange, the first speaker's question is not directly answered by the other speaker.

> "Do you want to have dinner tonight?"
> "I have an exam tomorrow."

The plain content of the reply would appear to violate the maxim of Relation ("Be relevant," Grice, 1975). And so the first speaker, upon hearing the reply, is left to infer the meaning that the replier intended to convey by assuming that there is some level at which the maxim is not being violated, even if it appears so at the surface (Levinson, 1983).[1]

[1] A distinction worth noting is that between conversational implicature and conventional implicature. A conversational implicature arises from the context within the conversation in which the utterance is made; in contrast, conventional implicature relies solely on the content of an utterance. A prototypical example of a conventional implicature is the sentence "The

There have been criticisms of Grice's (1975) argument (e.g., Sperber and Wilson (1986) argue that the maxims are so vague as to be unhelpful), but the fundamental point that utterances carry non-conventional meaning is generally accepted. Implicatures and indirect answers of this sort are very common in conversations—occurring in 27% of question/answer scenarios by one account (Rossen-Knill et al., 1997). It follows, then, that large language models trained and productized as chat systems would be more effective if able to use implicature. In addition, users used to human conversation are likely to interact with systems in a way that relies on the system correctly interpreting implicatures in their utterances, even if they do not deliberately set out to do so.

2.1 Prior Work on Implicature

In Louis et al. (2020), a model derived from BERT is trained to predict yes/no answers from a large corpus of indirect question/answer pairs. The authors found that this approach is largely successful, with an accuracy of 80%.

One of the first pieces of research looking at large language models'—rather than models trained specifically for this—ability in this regard is Zheng et al. (2021). The authors introduce a generated dataset of conversations containing implicatures, and then use it to evaluate a number of models' abilities. They note that the use of synthetic datasets if often criticized, and argue that any unnaturalness in their dataset is unrelated to implicatures, since they take care to use "pragmatic phenomena existing in daily conversations" (Zheng et al., 2021).

The BIG-bench benchmarking suite for language models also includes an implicature task (Maru and Bevilacqua, 2022). The authors use a dataset of natural implicatures produced by George and Mamidi (2020), avoiding one of the pitfalls of Zheng et al. (2021). However, Maru and Bevilacqua cut down the dataset by more than half, significantly limiting the size of their analysis.

Hu et al. (2023) look at language models' pragmatic abilities across a number of phenomena, including violations of the Gricean maxims. Performance at answering multiple-choice questions that rely on non-literal understanding is compared across a number of models and with human performance at the same task. They find that the best performing model tested (`text-davinci-002`) performs well above random chance, and often approaches human performance in those tasks. The authors use an expert-curated dataset consisting of 20–40 items per phenomenon. They note that, while this has the significant advantage of being a reliable dataset, its size is a limiting factor.

2.2 Ruis et al. Experiment

In Ruis et al. (2024), the authors look to evaluate the performance of a number of language models at recovering implicatures. They use a dataset of question/response pairs where the responses do not directly answer the question, but carry an implicature. Their experiment takes two forms: looking at the likelihood that the model predicts a 'yes' answer or a 'no' answer in response to an implicature, and a completion-based task where the models are instructed to generate text indicate whether the value of the implicature is yes or no.

For the likelihood task, they give the model a prompt that contains the question, the response, and then establishes a context in which it would be appropriate to output a yes/no answer. Determining whether the model has successfully recovered the correct value of the implicature is done by comparing the likelihoods assigned to the 'yes' and 'no' answers and checking whether the higher likelihood answer matches the implicature value from the dataset. This approach has the advantage of avoiding situations where, if used to generate text, the model would produce output that is neither 'yes' nor 'no,' which would prevent them from easily assessing the model's performance. This has the significant shortcoming, however, that not all models tested provide a way to access the likelihoods of the output. In particular, because some models—such as GPT-3.5-Turbo and GPT-4—are not publicly available (as is the case for a number of the additional models we test in Section 3), the experiments that can be conducted are limited to those that can make use of the online APIs that the developers elect to provide.

For the completion task, Ruis et al. use the same prompts but instead use the model to generate text. If the response ends with the words 'yes' or 'no,' then the responses is considered valid. It's considered correct if the yes/no response matches the

queen is English and therefore brave": the word *therefore* gives rise to the implication that being brave follows from being English (Davis, 2024). This example also highlights the pragmatic phenomenon of *presupposition* (it presupposes that there is currently an English queen), another pragmatic phenomenon that can have important implications (no pun intended!) in LLM-based work (Srikanth et al., 2024).

dataset's value for the implicature of that data point.

They also look at human performance at recovering implicatures in this data set. The same data is given to a group of human annotators who, through an online crowdsourcing platform, are instructed to finish each with 'yes' or 'no' based on what is contextually appropriate. The human annotators achieved an average accuracy of 86%.

Ruis et al. conducted this evaluation comprehensively with 17 different language models, divided into four categories (base models, dialogue fine-tuned, benchmark instruction-tuned, and example instruction-tuned), across 0-shot, 1-shot, and 5-shot scenarios. They find that the models in the Example IT category ("LLMs fine-tuned on tasks with natural instructions for each example," Ruis et al., 2024) consistently perform the best. They also find that, in certain circumstances, the best performing language model (GPT-4) achieves comparable accuracy to the human annotators.

3 Replication

Since Ruis et al. (2024) is one of the more comprehensive pieces of research on language models' performance with implicatures, we began looking into conversational implicature via a very standard approach: replicating the previous findings then seeing whether the results they obtained extend to newer models. We characterize this approach as "naïve" in the sense that it did not involve any particularly careful thought about the actual quality of the previous benchmark in terms of its data or task definition, nor were we particularly concerned with the specifics of the prompts used in the prior work. We simply took the previous benchmark on board uncritically and we assumed that, most likely, advances in language model size and general performance would give us updated baselines to beat.

Our attempt to replicate the results of Ruis et al. (2024) used the same data and a subset of the language models tested there. We also tested several newer models (GPT-4o, Google's Gemini 1.5 Pro, Anthropic's Claude 3, and Meta's Llama versions 3.2 and 3.3) and compared those results. We used the original Ruis et al. (2024) code, adapted for changes in some of the model vendors' APIs.[2] Because, as noted in Section 2.2, the APIs for GPT-3.5-Turbo and GPT-4 (among others) do not provide likelihood information, we only attempted to replicate the completion-based task.

3.1 Modifications

Closely related to prompt engineering, "answer engineering" refers to design choices that facilitate extraction of useful responses from LLM output (Schulhoff et al., 2024). We observed that some original prompts provided LLMs with too much latitude, e.g. "Finish the following text:" when the goal was a yes or no. In order to induce some of the language models (in particular, GPT-3.5-Turbo) to more reliably output yes/no responses as expected by the code, when asked in the 0-shot context for the value of an implicature, we minimally altered some of the prompt templates (see Appendix A): the three original templates which included "Finish the following text:" were modified to read "Finish the following text with yes or no:". This improves the yes/no format consistency of the output; we further modified the Ruis et al. (2024) code to identify the model's answer, not based on the last word of the output, but instead by checking if the response contains, as a whole word, 'yes' or 'no.'

The choice of models was based on those in Ruis et al.'s (2024) Example IT (instruction-tuning) category that were still available. The `text-davinci` models were deprecated by OpenAI in 2024 and are excluded here (OpenAI, 2023a). The Cohere-command-52B (`cohere-command-xlarge`) model is also no longer available; we used Cohere's Command R+ model. The code was extended to allow testing Google and Anthropic models using their APIs, as well as locally-run, open-source models via Ollama.

3.2 Results, Expected...

Table 1 shows the mean and standard deviation in accuracy across the different prompt templates for each of the models tested. For both of the original OpenAI models tested and for all k, accuracy has improved over Ruis et al.'s (2024) results. GPT-4 remains more accurate than GPT-3.5-Turbo though (and is comparable to GPT-4o). Our results also agree with Ruis et al. (2024) that moving from 0-shot to 1-shot to 5-shot does not consistently improve the models' performance.

It is difficult to identify the source of the improvements due to the generally closed nature of the model vendors. But, we expect that the change is likely due to ongoing refinement of the models. For instance, OpenAI notes that they regularly up-

[2] The code can be found on GitHub at `https://github.com/a-korde/llm-implicature-experiment`.

Model	0-Shot	1-shot	5-shot
GPT-3.5-Turbo[3]	77.4% ± 5.9	77.2% ± 4.5	77.6% ± 4.9
GPT-4	86.1% ± 0.7	83.3% ± 0.5	83.9% ± 0.3
GPT-4o	83.1% ± 4.8	84.2% ± 2.9	83.3% ± 2.5
Cohere Command R+	79.8% ± 3.9	80.3% ± 2.6	80.9% ± 1.6
Claude-3.5-Sonnet	**85.6% ± 1.6**	**88.1% ± 1.0**	**89.0% ± 0.6**
Gemini-1.5-Pro	83.5% ± 1.9	84.4% ± 4.2	83.8% ± 4.6
Llama-3.2-3B	60.9% ± 6.5	73.1% ± 13.0	69.9% ± 5.8
Llama-3.3-70B	84.2% ± 1.9	84.9% ± 1.8	84.9% ± 1.2

Table 1: The k-shot accuracy of a subset of the models tested in Ruis et al. (2024), as well as additional models, using our modified prompt templates (see Appendix A). Accuracy is averaged across the different prompt templates.

date models. When these tests were undertaken, the current versions of the OpenAI models used were `gpt-3.5-turbo-0125`, `gpt-4-0613`, and `gpt-4o-2024-08-06`. The Cohere model used was `command-r-plus-08-2024`. The Claude version used was `claude-3-5-sonnet-20241022`. The Gemini version used was `gemini-1.5-pro-002`.

3.3 ...And Unexpected

"The most exciting phrase to hear in science, the one that heralds new discoveries, is not 'Eureka!' but 'That's funny...' "
—Isaac Asimov

We were surprised to see that the one Cohere model tested here showed a dramatic improvement in the 0-shot task over the Cohere-command-52B model tested by Ruis et al., which achieved an accuracy of only 60.2% ± 5.2. One possible explanation for this change was the simple fact that we tested a different model. Changes from the previous Command-52B model's training data or process could have had an impact on its capability in this metric. It would have been fairly natural at this point simply to leave it at that, and move forward with Table 1 as our new baselines—and indeed we considered doing so.

However, Ruis et al.'s (2024) hypothesis about Cohere-command-52B's markedly worse performance on the 0-shot task as compared to the 1- and 5-shot tasks led us to think about an alternative explanation. They hypothesize that the poor 0-shot performance is "not due to a lack of implicature understanding, but due to a failure to calibrate the yes/no likelihoods without examples" (Ruis et al., 2024). That is, they argue the 1- and 5-shot examples serve to clarify the task format and "prime the model towards producing outputs and following the yes/no structure" (Ruis et al., 2024). If that is the case, then our altered prompts (see above) specifically asking for yes/no responses may have contributed to the improved performance.

To test this hypothesis, we re-ran the experiment on Command R+ using the original, unmodified prompt templates from Ruis et al. (2024). In this context, we found that Command R+ performed vastly worse than with our modified prompts. In the 0-shot case, Command R+ had a mean accuracy of just 50.8% ± 48.7 at correctly identifying the value of the implicature. This poor performance, and the very high variability, comes from differing behavior across prompt templates. In three of the original prompt templates—those that were unmodified in our experiment—the model performed in line with our results: it achieved an accuracy of 85.5%, 81.7%, and 72.0% for templates 0, 2, and 3 respectively. With the other three original prompt templates—the ones that we *did* modify—the model performed extraordinarily poorly, with the implicature accuracy varying from 0.8% to 1.5%. The completion accuracy metric (indicating what fraction of the model's generated completions an identifiable answer could be extracted from) shows the same pattern: each of the prompt templates that we did not have to modify all produced usable responses in greater than 98.5% of cases, and those templates that originally used "Finish the following text:" resulted in usable responses in no more than 2.5% of cases.

When given prompts with one example of the task, Command R+'s accuracy jumps to a more expected 73.4%±9.1. The "Finish the following text:" prompts remain somewhat worse performers than the others, however, scoring 62.3%, 65.2%, and 66.0% in implicature accuracy and 83.0%, 87.2%, and 90.8% in completion accuracy. Table 2 gives a

[3] The GPT-3.5-Turbo model is referred to as "ChatGPT" in Ruis et al. (2024).

breakdown of the individual prompt results across $k = 0, 1$ for each of the original and modified prompt templates.

4 Discussion

We viewed the results of our replication attempt as equivocal. On the one hand, we we were able to reproduce the results of Ruis et al. (2024). Frequently, including in our own work, that kind of replication success is sufficient to move on to the more interesting business of trying to build better models and improve the state of the art.

On the other hand, the Asimovian "that's funny" that emerged in our experimentation invited deeper consideration that, we suggest, is more valuable than the replication itself. This is where our discussion pivots from a conversation just about conversational implicature, *per se*, to a reconsideration of the "naïve" approach we took—an approach that is, we would argue, typical of widespread practice in current NLP research—building on a closer look at our replication attempt as a case study.

4.1 Datasets

We begin with data. The experiments here and in Ruis et al. (2024) use a dataset of implicatures in dialogue that have been manually annotated with the value of the implicatures (George and Mamidi, 2020). The data were obtained from two categories of sources: questions from an English language comprehension test (specifically, from free practice versions of the TOEFLS test (English Test Store)) and film scripts from the Internet Movie Script Database (IMSDb). That both of these sources are authored and not naturally occurring could present a difficulty: they may not be representative of how implicatures are used in natural conversation. Movie scripts, in particular, may also be a poor indicator of a model's performance, because the entire script may well have been included in the model's training data.

The dataset's authors also do not go into detail on the labeling process, only noting that "The annotation is done manually by undergraduate students of linguistics, whose primary language of instruction is English" (George and Mamidi, 2020). While the correct answers are provided for the language comprehension test, the same is not true of the entries from movie scripts, and the implicature values provided in the dataset are presumably the judgments of the aforementioned students.

The authors originally intended to crowdsource the dataset of implicatures—going so far as to design and conduct an experiment using an online crowdsourcing platform—but ultimately discarded the data noting that they "did not obtain high-quality dialogue data" (George and Mamidi, 2020). They conclude that the task they designed is somewhat ill-suited to crowdsourcing because it requires more imagination and is less mechanical than is common on crowdsourcing platforms.

This problem is not entirely resolved by using their chosen data sources, though. For instance, the dataset includes an entry with the following context and response utterances, and says that the implicature—the answer to the context question—is 'yes.'

> "Have you found another school for the children?"
> "We're still shopping around."

This does not align with our judgment: "still shopping around" implies that a suitable option has yet to be found. What's more, the dataset also contains entries that (again, in our judgment) simply do not contain implicatures. In the following example, the response appears to be a direct answer to the question (even though it does not contain the word 'yes' or 'no').

> "Did he ever fall back on a run?"
> "All the time, sir." (Sorkin, 1991)

These patterns show a potential issue in using the George and Mamidi (2020) dataset to evaluate models' performance at recovering implicatures. The BIG-bench implicature task uses the same dataset, but narrows it down to a greater extent—such as by "[d]iscarding factual errors in the original dataset" (Maru and Bevilacqua, 2022). This further constrained dataset may be useful in accurately identifying models' performance at implicature recovery, but of course comes at the expense of being even smaller. Additionally, there are a number of other datasets that could be used to similarly evaluate models' performance, however they are not without their own pitfalls.

The GRICE dataset is a collection of conversations involving implicatures and multiple-choice style questions, the correct answers to which depend on recovering the implicature (Zheng et al., 2021). Unlike the George and Mamidi (2020) dataset, Zheng et al. do not explicitly annotate the

Prompt	k	Implicature	Completion
Template 1	0	0.8%	2.5%
	1	62.3%	83.0%
Template 4	0	1.2%	1.7%
	1	65.2%	87.2%
Template 5	0	1.5%	2.3%
	1	66.0%	90.8%
Modified Template 1	0	79.3%	100.0%
	1	77.7%	100.0%
Modified Template 4	0	78.8%	100.0%
	1	78.0%	100.0%
Modified Template 5	0	80.3%	100.0%
	1	78.8%	100.0%

Table 2: Breakdown of Cohere Command R+ implicature and completion accuracy across the original "Finish the following text:" prompts from Ruis et al. (2024) and our modified prompts.

value of the implicature in each conversation, but instead only which of the multiple choice answers is correct. The GRICE dataset could be used in conjunction with the likelihood based approach used in Ruis et al. (2024) (see background in Section 2.2) by evaluating which of the multiple-choice answers the model predicts is most likely to appear. Because the data is programatically generated, however, this may exhibit the same issue of unnaturalness as in George and Mamidi (2020). In that regard, the variety of the GRICE data is rather limited: there are only four subtopics used to generate the conversations, which all follow a relatively simple conversational structure.

The dataset used in de Marneffe et al. (2010) provides a more natural source of implicature data. The authors sourced data from transcripts of interviews aired on CNN from 2000–2008 and the Switchboard corpus of telephone conversations (see Jurafsky et al., 1997). Labels were assigned based on the distribution of judgments of 30 Mechanical Turk workers for each of the dialogues. This may provide a higher quality source of data for evaluating implicature recovery performance, but it comes at the expense of being substantially smaller ($n = 224$).

One of the larger extant datasets is the Circa dataset, comprising 34,000+ pairs of crowdsourced questions and indirect answers (Louis et al., 2020). Both the questions and answers are crowdsourced. Labeling of the answers is also crowdsourced and divides the answers into yes/no categories (along with a split between certain/strong and uncertain/weak) as well as unsure and 'in the middle' (neither yes nor no) categories. The Louis et al. dataset seems promising as it is substantially larger than any of the others considered.

While the particular examples we discuss are specific to conversational implicature, they are illustrative of the potential issues that can arise when relying uncritically on existing datasets or benchmarks and using them to evaluate different models. The nature and quality of a particular dataset can play a significant role in a model's performance, and can risk presenting a distorted picture when attempting to make comparisons across models (let alone across datasets/benchmarks).

4.2 Prompt Sensitivity

Next, we turn to the issue of prompt sensitivity when it comes to cross-model comparisons and structured generation as a potential solution. Our experiment contributes further evidence to discussion in the literature regarding the danger of conceptualizing prompting as just another way of getting answers from a machine, comparable to the algorithms of prior generations. For example, Loya et al., 2023 found that GPT-3.5-Turbo's performance on a task conducted in prior research could be worsened or significantly improved with relatively minor alterations to the prompt. Our results in Section 3.3 reinforce the point: a difference of just four words ("with yes or no") dramatically changed the model's score on this benchmark. These observations suggest that the sensitivity of performance to prompt specifics is an essential consideration in any experiment using LLMs, and tools for evaluating prompt sensitivity (e.g., Sclar et al., 2024; Zhuo et al., 2024) should be a part of any future benchmark development process.

In terms of mitigating the risks of prompt sensitivity, Ruis et al. (2024) did so, to some extent, by using a set of six different prompts, rather than a single one. They divide the prompts into two groups: natural (prompts 1, 4, and 5) and structured (prompts 0, 2, and 3). However, as shown by the results with Command R+ (see Section 3.3), this was not entirely successful: Command R+ has consistent performance across prompts within a single group, but performs substantially differently between the natural prompts and the structured prompts.

In addition, chain-of-thought prompting (Wei et al., 2022), one of the techniques used by Loya et al., is also explored in Ruis et al. (2024). They found that 5-shot evaluation with chain-of-thought prompting brought GPT-4 to comparable performance to their human baseline. This improvement over the non-chain-of-thought results suggests that it is difficult—through completion tasks alone—to determine to what extent a language model has captured generalizations about implicatures.

Another way of avoiding the inherent prompt sensitivity of large language models is to avoid using text-generation tasks to study them. Instead, Ruis et al.'s (2024) comparing the relative likelihoods of multiple possible options would be more resilient to minor variations in the prompt. Unfortunately, the fact that state-of-the-art language models are developed by corporations that do not publish the full models presents a roadblock to studying them in more detail (e.g., OpenAI, 2023b, "Given both the competitive landscape and the safety implications of large-scale models like GPT-4, this report contains no further details."). Because access to the models is gated behind corporate APIs, which do not provide this information, research like ours is unable to use this technique.

Before we turn to structured output as a potential method for addressing the pitfalls we found associated with prompt sensitivity, we again emphasize that, although the specific issues discussed here depend on the use case and particular models under consideration, the broader issue of prompt sensitivity is fundamental to all large language models, including both closed source and open source (Sclar et al., 2024). As Errica et al. (2025) note, results from any model trained to maximize a likelihood objective are going to be sensitive to all features of the prompt that affect its probability.

4.3 Structured Output

In the interval between our original experimentation and writing this paper, structured output became an option for many LLMs: it is possible to make LLM text-generation requests explicitly defining the desired output format and limiting the model's output to that which conforms to the specified format. OpenAI's API now supports structured output by allowing the user to provide a JSON schema which the output must match (Pokrass, 2024): they describe a sampling process during text generation as "determin[ing] which tokens are valid to be produced next based on the previously generated tokens and the rules within the grammar that indicate which tokens are valid next." Ollama similarly supports providing a JSON schema to restrict the output (Ollama, 2024). Perhaps this renders many prompt sensitivity concerns moot?

We tested both GPT-4o and Llama 3.2 using a version of the Ruis et al. (2024) task adapted to use structured output. Rather than directly parsing the text, we used a JSON schema to have the model generate a JSON object containing a single boolean property representing the value of the implicature.

It turns out that, although structured output helps, LLMs persist in being inappropriately sensitive to details of the way they are called. In particular, note that in defining the JSON schema for the output, we were faced with the choice of what *name* to give to the boolean property representing the recovered value of the implicature. Although initially the grammar constrains the possible tokens to produce the JSON key, notice that, per the quote above, the key itself is part of the context and thus *the name of the key* will affect how the value for that property is generated.

To confirm this makes a difference, we tested both GPT-4o and Llama 3.2 using the original prompt templates from Ruis et al. (2024) using several different names for the boolean property, the results of which are shown in Table 3. We found that the property name can have a significant impact, though to what extent is variable.

Furthermore, we found that performance is still somewhat sensitive to the prompt, despite the constraints on the output. Table 4 shows the accuracy of Llama 3.2 for each prompt template in the structured output task. We note that adding "with yes or no" to prompt templates 1, 4, and 5 still produces a marked accuracy difference. That said, we also note that unmodified templates (0, 2, and 3) exhibit

JSON Key	GPT-4o	Llama 3.2
answer_is_yes	80.3% ± 6.0	60.2% ± 4.7
implicature_is_yes	80.2% ± 5.5	56.5% ± 2.5
implicature_value	70.2% ± 14.4	55.2% ± 3.1

Table 3: Mean accuracy across prompt templates for GPT-4o and Llama 3.2 depending on the key name in the JSON schema, when tested with the unmodified prompt templates and $k = 0$.

Prompt	Original	Modified
Template 0	64.0%	63.0%
Template 1	55.7%	60.8%
Template 2	65.0%	61.8%
Template 3	65.5%	65.0%
Template 4	55.0%	58.7%
Template 5	56.0%	61.2%

Table 4: Structured output accuracy for Llama 3.2 across the original and modified prompt templates (for templates 1, 4, and 5) when tested with answer_is_yes as the JSON key for $k = 0$.

a similar difference in some cases, so this effect may be within the run-to-run variance of the test.

Overall, we find that, although structured output may address the challenge of extracting information from LLM output, prompt sensitivity remains a significant concern. Put plainly: structured output affects the output's *structure*, not its substantive *content*. Instructions given to the model continue to have an impact on its apparent performance at a task, even if the model now always produces "grammatically correct" output. Additionally, structured output introduces the additional challenge of the output grammar itself (such as the names of the JSON keys) also affecting performance.

4.4 Other Paths Forward

As an alternative to seeking LLM-engineering solutions to the problems we are describing—something that in our view requires the efforts of the entire broader community—we conclude our discussion by considering underlying properties of the linguistic phenomenon being studied as a potentially more effective way to analyze the capabilities of language models. This can be thought of as a general strategy that we apply here to the specifics of conversational implicatures as a problem space.

Defeasibility and Reinforceability of Implicatures Two of the characteristic features of implicatures are that they are both defeasible and reinforceable (Levinson, 1983). They are defeasible in that the speaker of an implication-carrying utterance can defeat or cancel the implication in a subsequent utterance (for example, by saying something along the lines of, "But it's not actually the case that <*implication*>."). Similarly, they are reinforceable, and the speaker could emphasize what was previously implied. It's important to note that what makes the case of an implicature different from another utterance is that defeating or reinforcing an implication-carrying utterance neither produces a contradiction nor sounds redundant. By contrast, attempting to defeat an ordinary sentence does result in a contradiction and attempting to reinforce it often sounds redundant.[4]

Those differences could be used to test a model's sensitivity to implicature in a context where the likelihood of a string can be obtained from the model. By starting with a single question and an answer to it phrased both explicitly and as an implicature, and then comparing the likelihood of each of those being followed by a sentence that defeats/contradicts it, it may be possible to identify whether the model has recovered the implicature and the fact that it is an implicature. Flatly contradicting a prior sentence should be relatively unlikely. But, if the model has identified the implicature, then defeating it should be substantially more likely than the case of contradiction. Similarly, a sentence that repeats the same meaning as the previous one should be less likely in the case where the previous sentence is explicitly saying the same thing as compared to when the meaning of the previous sentence is provided by implication.

Unfortunately, this hypothesis is not readily testable at present, owing to the lack of likelihood information provided by the APIs for state-of-the-art language models.

Direct Inquiry vs. Conversation Continuations Our final observation is that evaluating language models' competence at recovering implicatures using a strategy of simply prompting them with instructions to evaluate the yes/no value of an implicature may not effectively represent their use of implicature in conversations. Presumably little of the models' training datasets consists of people directly asking what the meaning of an implication-carrying sentence is (aside, perhaps, from students of semantics or pragmatics). It is more likely that

[4] Levinson (1983) notes that there are circumstances, such as involving stress, where other types of sentences can be reinforced without issue. But those are not germane to our discussion.

the use of implicatures in the wild—and the conversations flowing therefrom—are better represented in the training data.

Since large language models are fundamentally constructed as text prediction/generation systems (e.g., "GPT-4 is a Transformer-style model pretrained to predict the next token in a document" OpenAI, 2023b), a task aimed at probing the same question but formulated to the context of text prediction/generation may produce more representative results. For example, given a context question and a response utterance carrying a conversational implicature, using a language model to generate a continuation of that conversation may provide another avenue for determining whether the model recovered the value of the implicature. If the model has recovered the value of the implicature, then the generated conversation should continue to flow naturally. If it has not, then there would be a break in the common ground and the conversation should be anomalous in some way.

5 Conclusions

With regard to conversational implicature, we have contributed an updated evaluation showing that Ruis et al.'s (2024) results hold up, improve with newer models, and that hoped-for improvements when moving from 0-shot to 1-shot to 5-shot in-context learning are not consistent. In addition, however, our simple attempt at replicating prior work using more up-to-date LLMs foregrounded deeper issues, ones that connect to broader questions about how to use and evaluate LLMs.

One key takeaway involves *data quality*, which receives little attention in NLP. In contrast to other fields like survey research and social sciences that have developed established, systematic frameworks for data quality assessment (Pipino et al., 2002; Groves and Lyberg, 2010; Birkenmaier et al., 2024), NLP research still largely lacks such frameworks and, despite some recognition of the problems (Bender and Friedman, 2018; Gebru et al., 2021; Northcutt et al., 2021) and emerging efforts to systematize data quality approaches (Dang and Verma, 2024; Mishra et al., 2020), there is scant evidence to suggest that common best practices are moving in that direction.

A second takeaway concerns the use of completion-based tasks. Our results and discussion suggest that completion-based tasks should be viewed with greater caution than they presently are, particularly for reasons associated with prompt sensitivity. Unfortunately, the constraints commercial LLM providers place on availability for alternatives, e.g. use of likelihoods, stymie otherwise potentially useful and creative solutions. We have suggested that in the absence of general solutions, finding ways to exploit relevant properties of the problem may be a better, or at least complementary, path forward.

A third takeaway concerns the pace of change in NLP. We attempted replication because models are constantly being updated. Having identified a problem with insufficiently constrained LLM output, we introduced solutions (e.g. prompt rephrasing)—only to find that by the time we were writing about the effort, still more recent developments in structured output capabilities required their own experimentation and evaluation, *and*, naturally, still did not fully fix the problem. Our takeway here is that the remarkably rapid change in NLP is both a blessing and a curse: in general we obtain better and better models and approaches, but there is barely any time to actually think deeply when so much effort is needed just to keep up. We would suggest that the field could benefit from a dose of slow science (Stengers, 2018), a perspective that de-emphasizes performance targets, deadlines, and market-based influences in favor of deeper thinking and curiosity-driven progress.

Finally, it is worth considering here, as with any attempt at creating an objective benchmark to measure the quality of a large language model, how the metric being used relates to the actual goal being pursued. Achieving a perfect score—or even a human-level score, like GPT-4—does not mean that a model has necessarily captured the same generalizations about implicatures that humans have. It may be that building or refining a model in order to improve its score on the Ruis et al. (2024) benchmark is not necessarily a productive way of improving its actual ability to *use* implicature. The broader take-away message is that we would do well to reminder ourselves regularly that "when a measure becomes a target, it ceases to be a good measure" (Goodhart's Law, Strathern, 1997).

Acknowledgments We are grateful to our anonymous reviewers for their thoughtful comments that improved the quality of the paper, Ruis et al. for their code and data, and the providers of the models that we used in our experimentation.

References

Emily M Bender and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604.

Lena Birkenmaier, Jessica Daikeler, Lisa Fröhling, Tobias Gummer, Clemens Lechner, Vanessa Lux, and Sebastian Ziaja. 2024. Defining and evaluating data quality for the social sciences: Position paper. GESIS Papers, 2024/06. Köln: GESIS – Leibniz-Institut für Sozialwissenschaften.

Viet Minh Hoang Dang and Rakesh M Verma. 2024. Data quality in nlp: Metrics and a comprehensive taxonomy. In *Advances in Intelligent Data Analysis XXII*, volume 14641 of *Lecture Notes in Computer Science*, pages 305–318. Springer.

Wayne Davis. 2024. Implicature. In Edward N. Zalta and Uri Nodelman, editors, *The Stanford Encyclopedia of Philosophy*, Spring 2024 edition. Metaphysics Research Lab, Stanford University.

Marie-Catherine de Marneffe, Christopher D. Manning, and Christopher Potts. 2010. “Was it good? It was provocative.” Learning the meaning of scalar adjectives. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 167–176, Uppsala, Sweden. Association for Computational Linguistics.

English Test Store. `https://englishteststore.net/`. Accessed: 2025-06-02. [link].

Federico Errica, Davide Sanvito, Giuseppe Siracusano, and Roberto Bifulco. 2025. What did I do wrong? quantifying LLMs’ sensitivity and consistency to prompt engineering. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1543–1558, Albuquerque, New Mexico. Association for Computational Linguistics.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. 2021. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92.

Elizabeth Jasmi George and Radhika Mamidi. 2020. Conversational implicatures in english dialogue: Annotated dataset. *Procedia Computer Science*, 171:2316–2323. Third International Conference on Computing and Network Communications (CoCoNet’19).

Herbert P Grice. 1975. Logic and conversation. In *Speech acts*, pages 41–58. Brill.

Robert M Groves and Lars Lyberg. 2010. Total survey error: Past, present, and future. *Public Opinion Quarterly*, 74(5):849–879.

Alexander Hoyle, Pranav Goel, Andrew Hian-Cheong, Denis Peskov, Jordan Boyd-Graber, and Philip Resnik. 2021. Is automated topic model evaluation broken? the incoherence of coherence. *Advances in neural information processing systems*, 34:2018–2033.

Jennifer Hu, Sammy Floyd, Olessia Jouravlev, Evelina Fedorenko, and Edward Gibson. 2023. A fine-grained comparison of pragmatic language understanding in humans and language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4194–4213, Toronto, Canada. Association for Computational Linguistics.

IMSDb. The internet movie script database. `https://imsdb.com/`. Accessed: 2025-06-02.

Daniel Jurafsky, Elizabeth Shriberg, and Debra Biasca. 1997. Switchboard SWBD-DAMSL shallow-discourse-function annotation coders manual, draft 13. Technical Report 97-02, University of Colorado, Boulder Institute of Cognitive Science, Boulder, CO.

Stephen C. Levinson. 1983. Conversational implicature. In *Pragmatics*, chapter 3. Cambridge University Press.

Annie Louis, Dan Roth, and Filip Radlinski. 2020. “I’d rather just go to bed”: Understanding indirect answers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7411–7425, Online. Association for Computational Linguistics.

Manikanta Loya, Divya Sinha, and Richard Futrell. 2023. Exploring the sensitivity of LLMs’ decision-making capabilities: Insights from prompt variations and hyperparameters. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3711–3716, Singapore. Association for Computational Linguistics.

Marco Maru and Michele Bevilacqua. 2022. Implicatures. README.md.

Swaroop Mishra, Anjana Arunkumar, Bhavdeep Singh Sachdeva, Chris Bryan, and Chitta Baral. 2020. Dqi: Measuring data quality in nlp. *arXiv preprint arXiv:2005.00816*. Available at: `https://arxiv.org/abs/2005.00816`.

Curtis Northcutt, Anish Athalye, and Jonas Mueller. 2021. Pervasive label errors in test sets destabilize machine learning benchmarks. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Ollama. 2024. Structured outputs. `https://ollama.com/blog/structured-outputs`. Accessed: 2025-06-02.

OpenAI. Models. `https://platform.openai.com/docs/models/gpt-4-turbo-and-gpt-4`. Accessed: 2025-06-02.

OpenAI. 2023a. GPT-4 API general availability and deprecation of older models in the completions API. https://openai.com/index/gpt-4-api-general-availability/. Accessed: 2025-06-02.

OpenAI. 2023b. GPT-4 technical report. *Preprint*, arXiv:2303.08774.

Leo L Pipino, Yang W Lee, and Richard Y Wang. 2002. Data quality assessment. *Communications of the ACM*, 45(4):211–218.

Michelle Pokrass. 2024. Introducing structured outputs in the api. https://openai.com/index/introducing-structured-outputs-in-the-api/. Accessed: 2025-06-02.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana. Association for Computational Linguistics.

Deborah Rossen-Knill, Beverly Spejewski, Beth Ann Hockey, Stephen Isard, and Matthew Stone. 1997. Yes/no questions and answers in the map task corpus.

Laura Ruis, Akbir Khan, Stella Biderman, Sara Hooker, Tim Rocktäschel, and Edward Grefenstette. 2024. The goldilocks of pragmatic understanding: fine-tuning strategy matters for implicature resolution by llms. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Sander Schulhoff, Michael Ilie, Nishant Balepur, Konstantine Kahadze, Amanda Liu, Chenglei Si, Yinheng Li, Aayush Gupta, HyoJung Han, Sevien Schulhoff, et al. 2024. The prompt report: A systematic survey of prompting techniques. *arXiv preprint arXiv:2406.06608*. Available at: https://arxiv.org/abs/2406.06608.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. *Preprint*, arXiv:2310.11324.

Aaron Sorkin. 1991. A few good men. https://imsdb.com/scripts/A-Few-Good-Men.html. Accessed: 2025-06-02.

Dan Sperber and Deirdre Wilson. 1986. *Relevance: Communication and Cognition*. Harvard University Press.

Neha Srikanth, Rupak Sarkar, Heran Mane, Elizabeth Aparicio, Quynh Nguyen, Rachel Rudinger, and Jordan Boyd-Graber. 2024. Pregnant questions: The importance of pragmatic awareness in maternal health question answering. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7253–7268, Mexico City, Mexico. Association for Computational Linguistics.

Isabelle Stengers. 2018. Another science is possible: A manifesto for slow science transl. *S Mueke Cambridge, UK: Polity (from French)*.

Marilyn Strathern. 1997. 'Improving ratings': audit in the British university system. *European Review*, 5(3):305–321.

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

Zilong Zheng, Shuwen Qiu, Lifeng Fan, Yixin Zhu, and Song-Chun Zhu. 2021. GRICE: A grammar-based dataset for recovering implicature and conversational reasoning. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2074–2085, Online. Association for Computational Linguistics.

Jingming Zhuo, Songyang Zhang, Xinyu Fang, Haodong Duan, Dahua Lin, and Kai Chen. 2024. ProSA: Assessing and understanding the prompt sensitivity of LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1950–1976, Miami, Florida, USA. Association for Computational Linguistics.

A Prompt Templates

Table 5 shows the original prompts from Ruis et al. (2024). Those that were modified in our experiment were changed to read "Finish the following text with yes or no:" for the instruction.

#	**Original template**	**Modified in our experiment**
0	Does the following response to the question imply yes or no? question: <*context utterance*> response: <*response utterance*> implicature:	No
1	Finish the following text: Esther asked "<*context utterance*>" and Juan responded "<*response utterance*>", which means	Yes
2	Is the implied meaning of the following response yes or no: question: <*context utterance*> response: <*response utterance*> meaning:	No
3	What is the intent of the following response, yes or no? question: <*context utterance*> response: <*response utterance*> intent:	No
4	Finish the following text: Karen asked "<*context utterance*>" and William responded "<*response utterance*>", which means	Yes
5	Finish the following text: Bob asked "<*context utterance*>" and Alice responded "<*response utterance*>", which means	Yes

Table 5: The prompt templates from Ruis et al. (2024) and whether they were modified in our experiment.