The Extractive-Abstractive Spectrum: Uncovering Verifiability Trade-offs in LLM Generations

Theodora Worledge **Tatsunori Hashimoto** **Carlos Guestrin**
Stanford Computer Science
worledge@stanford.edu

Abstract

Across all fields, respected experts cite their sources. While large language models (LLMs) excel at synthesizing information, they do not reliably cite sources, making it difficult to verify provided information. In contrast, search engines make sources readily accessible but burden users with synthesizing information. To rigorously examine the interplay between verifiability and utility, we build on classic theories of abstraction to propose an *extractive-abstractive spectrum* for cited generations; we define five operating points that span the extreme endpoints of *extractive* search engine outputs and *abstractive* LLM generations without reliable citation. We perform an analysis of the drawbacks and benefits of seven systems across the spectrum in the QA settings of search queries, information simplification, multi-hop reasoning, and health advice. As outputs become more abstractive, our human evaluations find that the proportion of properly cited sentences decreases by as much as 50%. Simultaneously, perceived utility improves by as much as 200% and users take up to 3 times as long to verify cited information. We recommend operating points for specific task settings and our failure analysis informs methods for citation identification, information retrieval, and multi-operating point systems.

1 Introduction

Large language models (LLMs) are increasingly capable of effectively answering questions (Ouyang et al., 2022; OpenAI et al., 2024). However, current LLMs lack accuracy in long-tail knowledge settings (Sun et al., 2023), align with unreliable sources[1] (Anagnostidis and Bulian, 2024), and disseminate bias and misinformation (Urman and Makhortykh, 2023). Therefore, it is critical to equip users in verifying LLM-generated claims, especially as users become more dependent on LLMs across different domains.

To achieve verifiability, generative search engines, including Google Gemini, Perplexity, and Microsoft Copilot, have combined LLMs with search engines to provide citations. However, these systems have poor citation quality; Liu et al. (2023) show that 25.5% of citations fail to support a claim they are expected to support and 48.5% of generated sentences requiring citation are not fully cited, across several widely-used deployed systems.[2] In scientific literature reviews researched and written by OpenAI's o3 DeepResearch and GPT-4.1, GPT-o3, Llama-4-Scout, and Gemini 2.5 Pro given web search capabilities, more than 50% of provided citations fail to support a claim they are expected to support and over 50% of generated sentences are not fully cited (Patel et al., 2025). Consequently, search engines remain appealing for users prioritizing verifiability. Search engine outputs are verifiable because users obtain information from sources directly but they sacrifice the fluency, concision, and synthesis of LLM responses; search engine users must open webpages, identify relevant information, and synthesize information across sources.

Our preliminary survey finds that users care about the differences between search engines and LLMs. Surveyed users prefer search engines to LLMs without citations, like ChatGPT, for answering high-stakes questions about health and taxes, while they prefer LLMs for research-intensive tasks with lower stakes, like trivia and explaining complex science to an inquisitive child

[1] BBC news article: "Glue pizza and eat rocks: Google AI search errors go viral"

[2] Liu et al. (2023) consider all sentences about the external world, including common knowledge, as verification-worthy. However, questions (e.g. "Would you like to learn more?") and self-referential statements (e.g. "I'm not sure.") are not considered to require citation.

Transactions of the Association for Computational Linguistics, vol. 14, pp. 2449–2474, 2026. https://doi.org/10.1162/TACL.a.811
Action Editor: Roi Reichart. Submission batch: 12/2025; Revision batch: 2/2026; Published 10/2026.
© 2026 Association for Computational Linguistics. Distributed under a CC-BY 4.0 license.

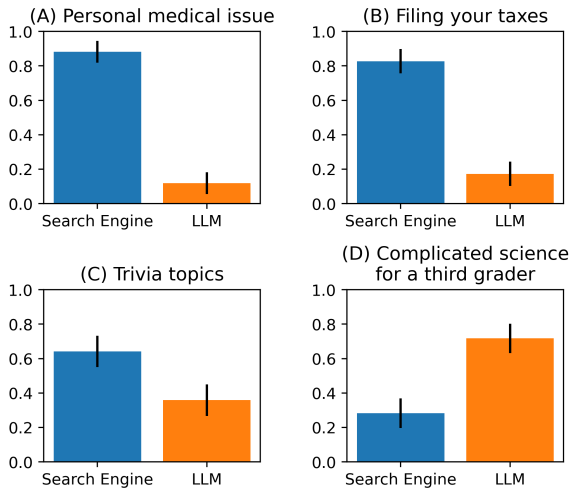


Figure 1: 200 users prefer to use search engines more for high-stakes topics, than for low-stakes queries. The error bars represent 95% confidence intervals.

(Figure 1). Furthermore, user preferences are informed by a lack of trust in LLMs. Users prefer search engines for the variety of sources, information provenance, and information reliability, in contrast to preferring LLMs for convenience and personalized answers (Figure 2A). There is an exciting opportunity to design trustworthy LLM systems preferred by users for low- and high-stakes queries alike.

In this work, we interrogate whether we can achieve verifiability while retaining the benefits of LLM responses by studying the *extractive-abstractive spectrum*. Search engines are *extractive* with respect to known sources in that they answer queries with snippets and urls of the sources themselves. Citations are inherent to search engine outputs. In contrast, LLMs are *abstractive* in that they synthesize content across disparate sources, often without providing citation. Our work defines five operating points that span the extractive-abstractive spectrum for cited generations and make distinct guarantees to users (Figure 2). For instance, the *quoted operating point* takes a step from extraction toward abstraction by extracting direct quotations from multiple sources and incorporating them into a single cited response. Another operating point is the *entailed generation*, where each sentence in the response is textually entailed by cited quotes from sources.

Despite intense scrutiny over citation systems (Nakano et al., 2021; Gao et al., 2023; Menick et al., 2022; Thoppilan et al., 2022; Slobodkin et al., 2024), existing work predominantly seeks to achieve a single operating point—the entailed generation. At the same time, quotation and paraphrase have received relatively little attention in cited generation systems, despite having long been quintessential in journalism and media (Gibson and Hester, 1999).

We perform careful human evaluations of seven different systems.[3] An ideal system provides helpful and fluent answers that are *verifiable*, i.e., entailed by citations that users can verify with minimal effort. To this end, we obtain human evaluations of answer quality, citation quality, and the complexity of verification. Our results show trade-offs in fluency, perceived utility, and verifiability—we observe as much as a 200% increase in perceived utility and 50% decrease in the proportion of properly cited sentences from extractive to abstractive systems, in addition to users taking up to three times as long to verify cited information. Popular deployed systems face these trade-offs as well; despite having high utility, 85.0% of verification-worthy sentences generated by the Google Gemini chatbot contain uncited information.[4] We also use a validated automatic evaluation pipeline to study trade-offs across the extractive-abstractive spectrum for three additional open and closed source models—we find the trends are robust.

We evaluate four query distributions that reflect real world question-answering (QA) tasks. These tasks include answering Google Search queries (Kwiatkowski et al., 2019), simplifying the style in which information is presented relative to the original sources (Jeblick et al., 2024; Choi et al., 2016), multi-hop reasoning (Ho et al., 2020), and providing medical advice (Zhu et al., 2020). We provide recommendations for specific applications informed by the extractive-abstractive trade-offs for these query distributions. We also identify failure modes that reveal opportunities in identifying citations after generation, task-specific information retrieval, and multi-operating point systems.

Our main contributions are: **(1)** We define the extractive-abstractive spectrum with five notable operating points that make distinct guarantees to

[3]Code and human evaluation data are made available here.

[4]We use Google's Vertex API to exclude sentences that are not of a factual nature, such as "Here is what I found.", and questions, such as "Would you like to learn more?", and find they are uncommon.

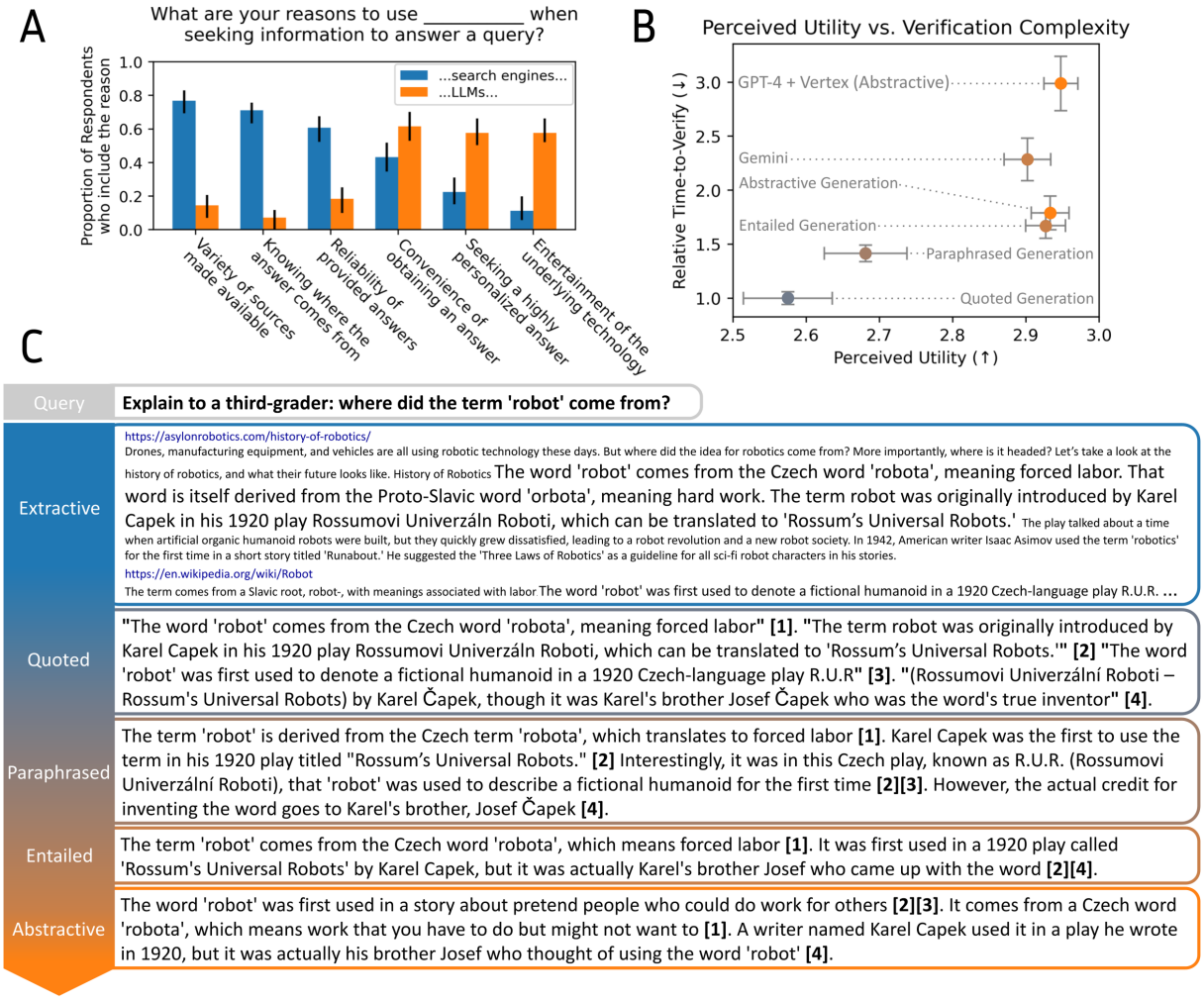


Figure 2: (A) User survey results regarding the reasons behind search engine and LLM preferences; search engines are desired for their ability to provide sources directly whereas LLMs are appreciated for their ability to synthesize information. (B) As generations become more abstractive, human evaluations of perceived utility increase with the relative time needed to evaluate whether sentences are properly supported by citation. (C) Examples of operating points across the extractive-abstractive spectrum from the reference implementations. While more extractive generations closely reflect the original source text (improving verifiability), more abstractive generations are more concise and customized to the query.

users. **(2)** We empirically characterize the trade-offs between generation quality and verifiability for four models and seven system implementations spanning the extractive-abstractive spectrum, using human and automatic evaluation. **(3)** We categorize and analyze the failure modes of cited LLM systems to inform directions for future work. **(4)** We recommend distinct operating points for different settings and discuss future work to improve verifiable LLM systems.

2 Cited Generation Formalization

We consider the task of cited generation for queries in the open-ended QA setting (example provided in Table 1). Given a user-provided natural language query, x, the systems we study produce a natural language generation y that contains inline citations to sources. The generation y is partitioned into n attributable units, $U = \{u_1, ..., u_n\}$, which our work defines as passages of text, often sentences. Attributable unit u_i is cited to m_i quotes, $Q_i = \{q_{i,1}, ..., q_{i,m_i}\}$, where $q_{i,j}$ is a direct quote from a known source. Accordingly, any claim contained within $q_{i,j}$ is interpreted in the context of its original source. We refer to the set of source texts from which quotes across all Q_i originate as S. Lastly, Q_i does not contain any cited quotes if u_i does not require citation. The cited source quotes Q_i, interpreted in their original source context, should support infor-

mation in u_i.

x	What is the diameter of Jupiter's largest moon?
y	$\underbrace{\text{Great question!}}_{u_1}$ $\underbrace{\text{Jupiter's largest moon,}}_{u_2}$ $\underbrace{\text{Ganymede, has a diameter of}}_{u_2}$ $\underbrace{\text{5,268 km}}_{u_2}$ [1][2].
Q_1	$\emptyset$
Q_2	[1] "Jupiter's moon Ganymede is the largest moon in our solar system, bigger than the planet Mercury and & dwarf planet Pluto." [2] "Ganymede has a diameter of 3270 miles (5,268 km) and is larger than the planet Mercury."
S	[1] Text of https://science.nasa.gov/jupiter/moons/ganymede/facts/ [2] Text of https://coolcosmos.ipac.caltech.edu/ask/201-What-is-the-largest- moon-in-the-Solar-System

Table 1: Sample input and outputs of the cited generation formalization.

Cited sources should be trustworthy and up-to-date. In this work, we assume that sources from the top Google search results meet these criteria within reason. Our work studies citation quality and relative perceived response utility, rather than absolute response accuracy. Applications that require high response accuracy should further filter internet search results using task-specific domain expertise for even higher quality sources.

3 The Extractive-Abstractive Spectrum

A text is *grounded* in sources if the sources semantically entail the text. Different texts may be grounded in the same sources with varying degrees of abstraction with respect to the sources. We believe these varying degrees of abstraction should be distinguishable to the human users for whom grounded generations are designed. However, previous approaches to framing and measuring abstraction are not necessarily intuitive to humans.

Early works on automatic summarization frame abstraction as an indicator of the coherence of generated text (McKeown et al., 1999; Knight and Marcu, 2002). Other works measure the abstraction of generated text with respect to sources using n-gram based metrics (Ladhak et al., 2022; Dreyer et al., 2023; Zhang et al., 2025b). However, neither coherence measures nor n-gram–based metrics capture meaningful differences in how generated text semantically relates to a source from a human reader's perspective. In contrast, Jing and McKeown (1999) study how humans use paraphrase and information removal to write abstractive summaries. Consistent with this approach, we define operating points (OPs) across the extractive-abstractive spectrum that correspond to different meaning relations between generations and cited source quotes. Our definitions correspond to distinct user experiences and also draw from work on meaning relations (Gold et al., 2019) applied to cited generations by Worledge et al. (2024). Figure 2C provides an example of each OP.

Extractive Output: An extractive output is a set of quotes from the sources. We refer to these quotes as snippets, as in Google Search. Citations are inherent to this OP. Formally, $\forall\ u_i,\ |Q_i| = 1,\ u_i = q_{i,1}$, where u_i may be multiple sentences.

Quoted Generation: A quoted generation achieves grounding by incorporating quotes of claims from known sources inline with unquoted words used sparingly to facilitate fluency. The quotes are taken *word-for-word* from the sources, are not misrepresented from the original context, and are clearly demarcated from non-quoted text. Formally, $\forall\ u_i,\ Q_i \Leftrightarrow u_i$ and every $q_{i,j}$ is a quoted substring of u_i.

Paraphrased Generation: A paraphrased generation rephrases information from cited source quotes. Each sentence conveys *the same information* as its cited source quotes; the generation does not remove any information, contract logical steps of reasoning, nor add new claims, relative to the cited quotes. Formally, $\forall\ u_i,\ Q_i \Leftrightarrow u_i$.

Entailed Generation: An entailed generation achieves grounding by rephrasing, removing, and logically transforming information relative to cited source quotes. Each sentence conveys *no more information* than its cited source quotes; the generation may remove extraneous information and contract logical steps of reasoning, but does not add new claims. Formally, $\forall\ u_i,\ Q_i \Rightarrow u_i$.

Abstractive Generation: Each sentence of an abstractive generation may contain uncited claims ($u_{i,\text{uncited claims}}$) and claims that are cited ($u_{i,\text{cited claims}}$). A sentence may convey *more in-*

formation than the cited source quotes. Formally, if $u_{i,\text{cited claims}} \neq \emptyset$, then $Q_i \Rightarrow u_{i,\text{cited claims}}$.

The OPs correspond to familiar user experiences or bridge the gap between them. Search engines provide extractive responses. LLM generations are abstractive, where there are no cited claims ($u_{i,\text{cited claims}} = \emptyset$). We speculate that generative search engines implement the entailed or abstractive OPs. Most prior work implements the entailed OPs (Section 10.1).

4 Cited Generation Implementations

We design comparable reference OP implementations for studying the extractive-abstractive spectrum by grounding responses to a query in the same source quotes. By grounding each OP generation for a query in the same quotes, we ensure that performance differences reflect the OPs instead of variations in information.[5] We implement four versions of the OPs below with GPT-4, GPT-5, Sonnet 4.5, and DeepSeek-V3.2, each hereafter referred to as "the model". We disable web search tools for all model calls and validate that our OP implementations are competitive with alternate implementations in Appendix A.3.5.

The **quoted generation** is generated from the model with few-shot examples, the query, and source snippets provided in-context. We denote this implementation as $\mathcal{M}_{\text{quoted}}(x,\ S)$. We prompt the model to respond to the query using word-for-word quotes from the snippets. Citations are identified by string-matching generated quotes to snippets. The **extractive output** is the list of snippets that contain the quotes in the quoted generation.

The **paraphrased**, **entailed**, and **abstractive generations** are revisions of the quoted generation; we prompt the model with few-shot examples, the query, and the corresponding quoted generation. The paraphrased implementation is $\mathcal{M}_{\text{paraphrased}}(x,\ \mathcal{M}_{\text{quoted}}(x,\ S))$, with the same function signature applying to $\mathcal{M}_{\text{entailed}}$ and $\mathcal{M}_{\text{abstractive}}$. The paraphrased prompt requests a rewrite, the entailed prompt requests a more concise generation, and the abstractive prompt permits the model to add relevant information not found in $\mathcal{M}_{\text{quoted}}$. For each sentence, few-shot prompting with the model is used to cite quotes from the quoted generation. Quotes are then cited within their original snippet.

[5] We hope to see future implementations that optimize individual OPs.

We also study two publicly available methods. The **GPT-4 + Vertex generation** is another instantiation of the abstractive OP. The response to a query is generated using GPT-4 without in-context sources. Using Google's Vertex API, we find citations for generated sentences from snippets retrieved from Google.[6] We refer to these citations as *post hoc* because the cited sources are not used in response generation and are instead retrieved using the generation itself. Our two implementations of the abstractive OP reflect model generations without cited generation constraints. Because we do not control the quantity of uncited information in abstractive generations across different models, abstractive results are not necessarily comparable across different models.

Lastly, we evaluate **Google Gemini** as a widely-used deployed system baseline. Like our implementations, Gemini cites sentences within sources, unlike other deployed systems that cite sources. While Gemini conditions generations on retrieved sources, some citations are post hoc.[7,8]

5 Human Evaluation

We conduct human evaluations of the GPT-4 reference, GPT-4 + Vertex, and Gemini generations for five quality and verifiability metrics over four query distributions. The exact questions and instructions shown to annotators are available in the project's Github repository.

5.1 Metrics

An ideal cited generation addresses the query fluently, effectively, and verifiably. We remove citation markers to evaluate **fluency**, the extent to which a generation is grammatically correct, coherent, and has smooth transitions, and **perceived utility**, the extent to which a user believes the generation concisely addresses the query in an appropriate style. Both are rated on three-point scales, three being high quality.

We measure verifiability with the precision of each citation, the citation coverage of each sentence, and the time to verify each sentence. First, we exclude sentences that do not require citation according to Google's Vertex check grounding

[6] "Google Vertex AI Check Grounding"

[7] "Google Cloud expands grounding capabilities on Vertex AI"

[8] "View related sources & double-check responses from Gemini Apps"

API.[9] The Vertex API indicates whether each sentence in a response contains information of a factual nature, including common knowledge. Self-referential statements and questions are not considered by the API to require grounding.[10]

Citation precision is the proportion of citations that support at least one claim in the sentence they follow. An imprecise citation is either irrelevant to or disagrees with its corresponding sentence. In Table 1, say u_2 was accompanied by a third citation: [3] "Ganymede has mountains, valleys, craters and old lava flows." Then, citation [3] would not be *precise* because the cited information is irrelevant to u_2.

Citation coverage is the proportion of sentences that are fully supported by their cited source quotes. In Table 1, say citation [2] did not exist. Then, u_2 would not have complete *citation coverage* because its only cited quote does not entail the claim that Ganymede's diameter is 5,268 km. Precision and coverage are judged by the intuitive test for meaning-level consistency where cited quote(s), Q_i, supports claim(s), u_i, if the annotator believes it is truthful to say, "According to Q_i, u_i" (Rashkin et al., 2023). We leave the task of identifying claims within a sentence to the discretion of the human annotators and their experience as information-seeking users.

Time-to-verify (T2V) is the wall clock time it takes a human to determine the coverage of a cited sentence. We define T2V for coverage because it reflects the user experience of ensuring that all information is supported by citations, unlike perfect precision which permits uncited information. To account for variable reading speeds, we report *relative T2V*; each T2V value is normalized by its annotator's average T2V from quoted generations.

5.2 Query Distributions

We obtain human evaluations of seven cited generations for 480 open-ended queries from four information-seeking QA distributions. We use retrieved sources from Google for NQ and Eta3G and gold-standard sources, i.e. sources with answers, provided by 2WikiMH and MASH.

Natural Questions (NQ) (Kwiatkowski et al., 2019) is composed of real Google Search queries. **Explain to a Third Grader (Eta3G)** is a variant of NQ where queries are prepended with "Explain to a third-grader:". Text written for a third-grade level relies more on simple and compound sentences, commonplace vocabulary, and explicit reasoning with fewer inferences than text written for mature readers. Accordingly, many sources from the internet are not written at a third-grade level and may contain advanced topic material.[11]

Answers to queries from **Wikipedia and Wikidata Multi-Hop (2WikiMH)** (Ho et al., 2020) combine premises from multiple sources to draw conclusions not present in any individual source. The queries are about entity relations e.g. "Who is the paternal grandfather of Prince Alexander Of Imereti?" Lastly, queries from **Multiple Answer Spans Healthcare (MASH)** (Zhu et al., 2020) seek medical advice, a setting where misinformation has high-stakes consequences.

5.3 Evaluation Procedure

Thirty-one Amazon Mechanical Turk (MTurk) annotators were admitted to the study after passing a screener and receiving personalized feedback. Throughout the study, we maintained an active presence through email and a dedicated channel on the Turker Nation Slack workspace.

Each annotator evaluated an equal number of each reference OP and an equal number of GPT-4 + Vertex and Gemini generations. To avoid familiarization with the queries and sources, no annotator evaluated the same query twice. We provide further details, inter-annotator agreement rates, and results from a pilot study performed before the MTurk evaluations in Appendix A.2.

6 Results and Analysis

6.1 Results across all query distributions

Fluency and perceived utility increase as generations become more abstractive (Figure 3). The quoted OP significantly improves fluency from the extractive output and fluency further benefits from the flexibility of the more abstractive OPs. The quoted OP significantly improves perceived utility from the extractive snippets by discarding irrelevant content. While the paraphrased OP further improves perceived utility, a more significant increase is achieved by the entailed OP. This increase is likely because both paraphrased and en-

[9] "Google Vertex AI Check Grounding"

[10] Examples of statements flagged as not requiring grounding by the API include "I'm unable to help, as I am only a language model" and "Here is what I found."

[11] Examples of text written for a third-grade reading level can be viewed on Khan Academy's 3rd Grade Reading & Vocabulary Course.

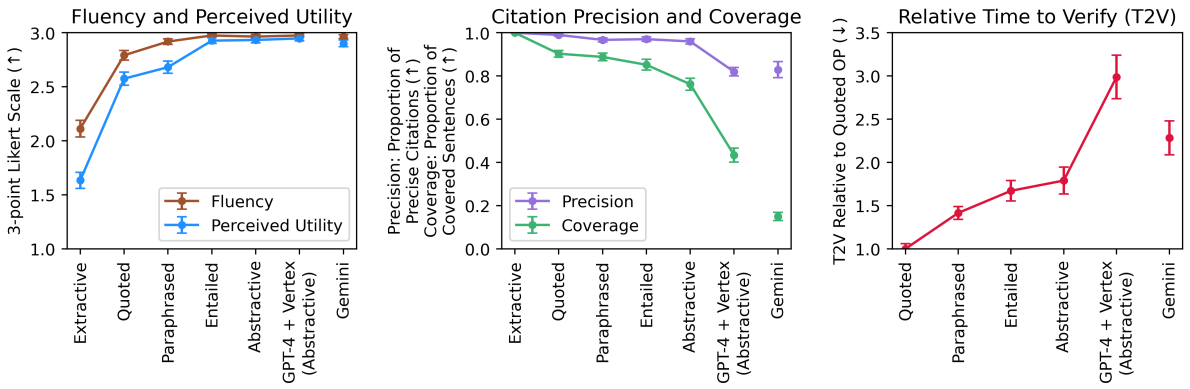


Figure 3: Human evaluation results averaged over the four query distributions. Fluency and perceived utility increase with abstraction, while citation precision and coverage decrease. Annotators take longer to evaluate coverage as generations become more abstractive. The error bars represent 95% confidence intervals.

tailed generations customize generated language to the query, but only the entailed generations can remove information and draw conclusions.

In contrast, citation precision and coverage degrade as generations become more abstractive (Figure 3). Citation precision is relatively similar (95%-100%) across the reference OP generations, but is significantly lower for the GPT-4 + Vertex and Gemini generations. These results indicate that conditioning response generation on predetermined source quotes improves citation precision, relative to the post hoc citation approach used in the GPT-4 + Vertex generations and likely employed by Gemini. While coverage drops significantly for the reference abstractive generations, which occurs by definition, coverage is even lower for GPT-4 + Vertex and Gemini generations, at 43.4% and 15.0% respectively.

Humans take longer to verify cited sentences for more abstractive generations (Figure 3). The quoted OP significantly expedites human verification; rewording quoted generations to be paraphrased increases T2V by over 40%. Some abstractive generations take up to three times as long to verify. The trend in T2V is not solely driven by per-sentence citation count (Appendix Table 8). Absolute T2V, T2V over covered sentences, and T2V over entire responses also trend upward with abstraction (Appendix Figures 8 and 7). We observe that the model's overall coverage score roughly correlates with T2V; it is possible that users take more time verifying information when they observe more frequent citation failures.

Gemini generations are dominated by entailed generations on citation precision, coverage, and relative T2V. Despite similar fluency and perceived utility, Gemini exhibits over 80% lower coverage and over 14% lower citation precision, with humans taking significantly longer to verify its cited sentences (Figure 3). Gemini's exceedingly low coverage (15.0%) suggests that cost and latency may hinder coverage in deployed systems. More broadly, high perceived utility paired with low coverage is concerning: responses seen as helpful but lacking citations may lead users to accept information without relevant grounding, potentially increasing the risk of misinformed decision-making.

6.2 Results by query distribution

For each query distribution, perceived utility increases, fluency increases, citation precision and coverage decrease, and T2V increases with generation abstraction (Appendix Figure 9). Perceived utility stagnates between the quoted and paraphrased OPs for 2WikiMH and MASH queries. Section 8.1 provides insight into these perceived utility failures by query distribution.

7 Automatic Evaluations and Additional Model Evaluations

While human evaluation is ideal and reflects user experiences with system shortcomings, we use automatic evaluation to extend our analysis to generations from additional models. Human evaluations are resource intensive and training crowd-sourced workers is logistically challenging. In order to determine whether trade-offs across operating points hold for generations from models besides GPT-4, we designed and validated automatic evaluation of

fluency, perceived utility, citation precision, and coverage. We find that evaluations from GPT-5 strongly correlate with human evaluations for each metric (Table 2). We do not include an automated T2V evaluation because T2V reflects distinctly human behaviors such as cognitive fatigue.

Metric	Correlation (r)	p-value
Fluency	0.962	0.003
Utility	0.965	0.010
Coverage	0.977	0.003
Precision	0.949	0.004

Table 2: The Pearson correlation coefficients between human and automated evaluations show that automated evaluations of our metrics for generation and citation quality are reasonable substitutes for human evaluation. Correlation is measured across the average human evaluation and average automated evaluation paired by cited generation type. The p-values correspond to a null distribution where the average evaluations are randomly permuted, rather than paired by cited generation type, and are calculated with a finite sample correction.

Using automatic evaluation, we show that the operating points generated by GPT-5, Claude Sonnet 4.5, and DeepSeek-V3.2 are subject to the same trade-offs across response and citation quality as GPT-4 generations (Figure 4). As fluency and perceived utility increase with abstraction, citation precision and coverage trend downward.

Coverage results for the abstractive operating point are not comparable across models; we do not control the extent of uncited information across abstractive implementations (Appendix Example A.1). However, abstractive generations are comparable to other generations from the same model; abstractive generations are model responses created without the constraint of citing information.

8 Failure Analysis

The inherent properties of the operating points lead to unique failure modes. We categorize and report the frequency of observed failure modes for fluency, utility, precision, and coverage by operating point for GPT-4 generations. By understanding the failures flagged by human reviewers, we provide insight into the strengths and weaknesses of each operating point from the user's perspective for practitioners and future work (Section 9.2).

Example 8.1: Abstraction improves fluency

ETa3G Query: Explain to a third-grader: when does the next allegiant movie come out

Quoted Generation: "The Divergent Series: Allegiant – Part 1" [1] was released on March 18, 2016, and there will not be a next movie as "Ascendent, but the film never saw the light of day" [2].

Paraphrased Generation: The movie "The Divergent Series: Allegiant – Part 1" came out on March 18, 2016 [1]. There won't be another movie called "Ascendent" because it was never made [2].

8.1 Fluency and utility failures

Most fluency and utility failures (any generation without the highest rating) in quoted generations are addressed by their paraphrased and entailed counterparts. Accordingly, we discuss fluency and utility failure modes of the quoted OP. Appendix Table 7 reports failure frequency by type.

Example 8.2: Concision improves with abstraction

MASH Query: How often should I check my blood sugar if I have diabetes?

Quoted Generation: "When you have diabetes, you may need to check your blood sugar throughout the day" [1]. "Give yourself a blood sugar test as often as your doctor advises" [2]. "To do it, you use a gadget that pricks your finger with a tiny needle" [3]. "You'll put a drop of blood onto a test strip" [4]. "The strip goes into a handheld device that measures your blood sugar level" [5]. "A1c Test: This is a blood test you'll get in your doctor's office at least twice a year, or as often as he recommends" [7]. "The results show your average blood sugar control for the past 2 to 3 months" [8].

Entailed Generation: If you have diabetes, check your blood sugar as frequently as recommended by your doctor using a blood glucose meter, which involves pricking your finger for a blood sample [1][2][3]. Additionally, undergo an A1c test at least twice a year to monitor your average blood sugar over the past few months [7][8].

Disfluent quote integration: Incorrect grammar arising from the clumsy integration of inline quotes frequently leads to fluency failures, even when the original sources themselves are grammatically correct. This failure is often corrected in the paraphrased output (Example 8.1). An-

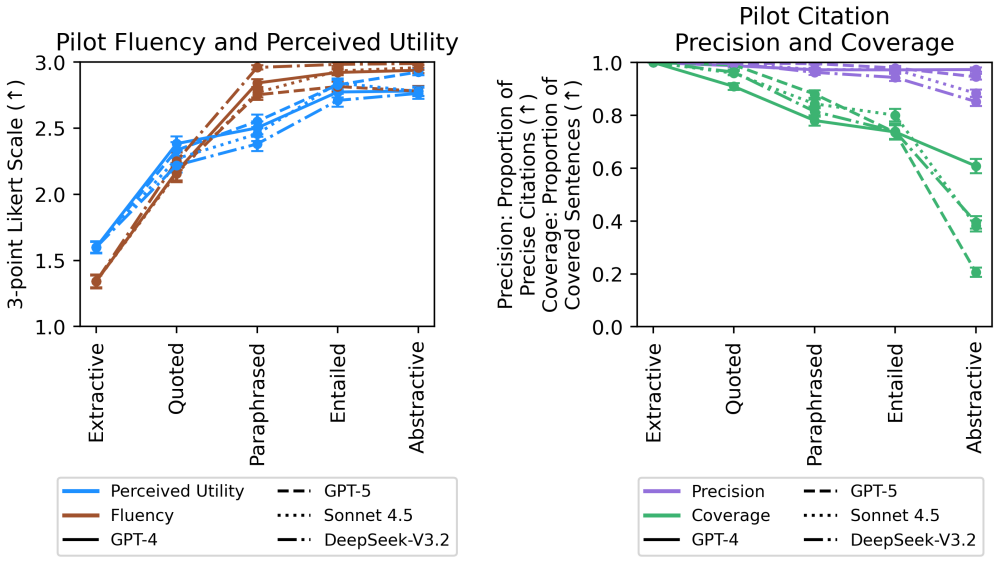


Figure 4: Automatic evaluation averaged over the four query distributions. Fluency and perceived utility increase with abstraction, while citation precision and coverage decrease. The error bars represent 95% confidence intervals.

other common fluency failure of quoted generation is the abrupt transition between sentences, especially when quotes originally written for different sources are used in the same generation.

Verbose: Quoted generations containing irrelevant information occur for all query distributions. For instance, 80% of MASH perceived utility failures contain unnecessary information, perhaps due to the use of gold sources with a plethora of tangentially topical information. Example 8.2 shows a quoted generation that includes extensive information about *how* to perform a medical test despite the question asking how *often* to seek the medical test; the entailed generation removes irrelevant information about how to perform the test. On average, the entailed generations are shorter than both the paraphrased and quoted generations (Appendix Table 8).

Premises without the conclusion: A low-utility generation may only enumerate premises without drawing the query's sought-after conclusion. This failure is most common for 2WikiMH quoted queries where conclusions are not explicitly stated in the sources. The quoted and the paraphrased generations in Example 8.3 only enumerate premises to address the query, while the entailed generation presents the desired conclusion, "No".

Example 8.3: Entailment enables reasoning

2WikiMH Query: Do both films La Vuelta De Martín Fierro and Joy Division (2006 Film) have the directors from the same country?

Quoted Generation: "La Vuelta de Martín Fierro is a 1974 Argentine film directed by Enrique Dawi" [1]. "Enrique Dawi(1927- 1988) was an Argentine film director and screenplay writer" [2]. "Joy Division is a 2006 British-German- Hungarian film directed by Reg Traviss" [3]. "Reginald Stephen' Reg' Traviss(born 12 February 1977) is a British film director and writer" [4].

Paraphrased Generation: 'La Vuelta de Martín Fierro,' a film from Argentina released in 1974, was directed by Enrique Dawi [1]. Enrique Dawi (1927-1988) was recognized for his contributions to Argentine cinema as both a director and a screenwriter [1][2]. On the other hand, 'Joy Division,' a film produced in collaboration between Britain, Germany, and Hungary in 2006, was helmed by Reg Traviss [3][4]. Reginald Stephen 'Reg' Traviss, born on February 12, 1977, is known for his work in the British film industry as both a director and a screenwriter [3][4]

Entailed Generation: No, 'La Vuelta de Martín Fierro' was directed by Argentine Enrique Dawi, while 'Joy Division' was directed by British filmmaker Reg Traviss [1][3][4].

Unsuitable style: Quoted generations with low utility may lack the style required by the query.

This failure mode is most common in Eta3G quoted generations; relatively few internet sources are written for third-graders. Example 8.4 shows a quoted generation that uses challenging vocabulary, such as "bicameral", "interim", and "synodical". In contrast, the paraphrased generation simplifies vocabulary and the entailed generation further simplifies the presented information.

Example 8.4: Entailment enhances style change

ETa3G Query: Explain to a third-grader: what is the hierarchy of the episcopal church

Quoted Generation: "In contrast, the Episcopal Church puts an emphasis on shared authority" [1]. "It operates under a polity that includes both episcopal and synodical governance" [2]. "The Episcopal Church is governed by a bicameral General Convention, which meets every three years, and by an Executive Council during interim years" [3]. "The General Convention consists of the House of Bishops and the House of Deputies" [4].

Paraphrased Generation: The Episcopal Church focuses on everyone having a say in how things are run [1]. It uses a system that combines leaders making decisions and groups of people coming together to decide things [2]. This church is led by a big meeting called the General Convention that happens every three years, and by a group called the Executive Council in the years in between [3]. At the General Convention, there are two main groups: one is made up of bishops, and the other is made up of deputies [4].

Entailed Generation: The Episcopal Church is like a big team that makes decisions together [1]. It has leaders called bishops and other members who meet every three years to decide on important things [3]. They work together to make sure everyone agrees [1].

Failure to interpret sources in-context: Sometimes, quoted generations fail to quote the answer from in-context sources. This failure mode affects any response conditioned on in-context sources and is more common if sources are retrieved, rather than gold-standard. In Example 8.5, the retrieved sources contain the answer, but the quoted generation fails to identify it. The abstractive generation uses parametric knowledge to produce the answer. A similar failure occurs if information retrieval fails to find sources that address the query.

Example 8.5

NQ Query: who plays jack's mom in 30 rock 2

Quoted Generation: "From 2007 to 2012, she had a recurring role as Colleen Donaghy on the NBC sitcom 30 Rock, a role that won her a third Emmy in 2007" [1].

Abstractive Generation: Elaine Stritch played Jack Donaghy's mother, Colleen Donaghy, on the NBC sitcom "30 Rock" from 2007 to 2012, a role that earned her an Emmy in 2007 [1].

Source: https://www.vulture.com/20...
Tonight's 30 Rock contained a few things [...] Jack's harpy mom, Colleen, played for seven seasons by the great Elaine Stritch [...]

8.2 Citation precision failures

We categorize different types of citation precision failures for the quoted, entailed, GPT-4 + Vertex, and Gemini generations. Appendix Table 5 contains the relative frequency of each failure mode.

Claim taken out of context: Even if the cited portion of a source appears to provide support for a claim, other parts of the source may indicate otherwise. This is the only precision failure of quoted generations because cited quotes can only be misrepresented by being taken out of context (Zhang et al., 2023). The quoted generation in Example 8.6 quotes an answer option for a multiple choice question, however, the quote is not presented as the truth in the context of the source.

Example 8.6: Quote taken out of context

NQ Query: right to property according to the constitution of india is a

Quoted Generation: "Right to property is legal right but not a Fundamental Right" [1].

Source 1:
https://www.toppr.com/ask/question/the-rig...
Consider the following statements [...]: 1. [1] Right to property is legal right but not a Fundamental Right. 2. Article 300 A was inserted in the Constitution of India by the 44th Constitutional Amendment. Which of the statements given above is/are correct

Keywords mistaken for entailment: Some precision errors are due to misjudging a semantically similar quote as entailing the target sentence. This error is common in the GPT-4 + Vertex (67.6%) and Gemini (75.0%) precision failures, likely because the citations are post hoc. This error is less frequent for quoted (0%) and en-

tailed (34.8%) precision failures, which generate responses from cited quotes. In Example 8.7, citation 3 is about the International Organization for Standardization language (ISO), rather than plain English, likely due to their similar function.

Example 8.7: Citation only contains relevant words

NQ Query: explain what is meant by plain english and why is it used

GPT-4 + Vertex Generation: Plain English is a style of writing that uses simple, straightforward language to communicate clearly [0]. It is used to ensure that information is accessible and easily understood by a wide audience [1][2][3].

Source 1:
https://www.iso.org/ISO-house-style.html
[...] writers adopt different approaches. [3] The priority for ISO documents is to be easily understandable and unambiguous. Use [...]

Supporting claim is highly relevant, but not specific enough: Some precision errors result from citing a source quote that makes a broader claim than the generation. This error is most common for entailed precision failures (60.9%) because rewording the source quote can distort the original claim to be more specific. In Example 8.8, the generation specifies the "coxal bone" but citation 1 discusses "the hip", a more general term.

Example 8.8: Cited source quote highly relevant yet misses a detail

NQ Query: the joint between a coxal bone of the pelvis and the sacrum

GPT-4 + Vertex Generation: The joint between a coxal bone of the pelvis and the sacrum is known as the sacroiliac joint [1]. This joint plays a crucial role in transferring weight and movement between the upper body and legs.

Source 1: https://www.spine-health.com/...
[1] The unique sacroiliac articulation between the sacrum and the hip provides a transition between the spine and the pelvis. While [...]

8.3 Citation coverage failures

A sentence lacking citation coverage either lacks citation completely, has precise citation only *partially* covering the sentence, or only has imprecise citation. The former two failure modes are the majority of coverage failures; Appendix Table 6 shows the prevalence of these three failure modes.

No citations: A complete lack of citation is the most common case of the GPT-4 + Vertex (67.7%), and Gemini (83.8%) precision failures, underscoring the challenges of identifying post hoc citations. This failure mode is not as common for the reference abstractive generations.

Partial coverage: Some coverage errors result from precise citations that do not support all claims in the sentence. In contrast to quoted (26.1%), the majority of paraphrased (58.6%) and entailed (67.0%) coverage failures result from partial coverage. Increasingly abstractive generations often synthesize information across more sources, requiring more citations for coverage.

9 Discussion

The trade-offs across the extractive-abstractive spectrum analyzed in Section 6 provide insight for domain-specific system design choices. We also recommend directions for future work informed by the our failure analysis (Section 8).

9.1 Optimal operating point differs across tasks

Due to the observed trade-offs across the extractive-abstractive spectrum, there is no one-size-fits-all OP across applications. The trade-offs motivate a segmentation of QA settings into three categories discussed below. Model developers should consider the priorities of their application area and implement favorable OPs.

In the first setting, high-stakes queries require searching a refined corpus for information already written in an appropriate form, but dispersed across sources. LLMs can help process retrieval queries and generate responses that stitch together source quotes. Due to the importance of verifiability, these settings are best served by extractive and quoted generations. LLM tools that assist lawyers in research for legal writing exemplify this setting. Errors in legal documents have serious consequences, including suspension of legal licenses and reputational harm.[12,13] Because verifiability is critical, current tools include Harvey's LLM-based instantiation of the extractive OP for law firms.[14] With lawyers formulating the legal arguments, more abstractive OPs may be unnecessary and risk unverifiable information.

[12] https://www.cbsnews.com/colorado/news/colorado-lawyer-artificial-intelligence-suspension/

[13] https://www.cbsnews.com/news/chatgpt-judge-fines-lawyers-who-used-ai/

[14] https://www.harvey.ai/

The second setting requires reformulating information from original sources into a different style, without a need for verifiability. In fact, tasks like creative writing (Mirowski et al., 2023) and brainstorming (Shaer et al., 2024) require novel content not necessarily found in existing sources; only the abstractive OP can achieve this goal.

In the third setting, queries require reformulating information across sources into a different style while providing high verifiability. This is the most challenging setting; no OP is optimal for both generation quality and verifiability. Tasks such as transforming clinical patient dialogue into medical notes (Biswas and Talukdar, 2024; Leong et al., 2024) require grounding in the original dialogue and substantive style change from layperson diction to medical terminology. Similarly, tasks such as simplifying patient medical records for layperson understanding (Jeblick et al., 2024; Choi et al., 2016) and simplifying legal contracts for lower reading levels (Ali et al., 2024) also require high verifiability and style change. The paraphrased and entailed generations can provide a balance between these goals. Regardless, the consequences of an error require experts to verify generations, underscoring the importance of low-complexity verification (Ali et al., 2024; Mathews, 2024; Jeblick et al., 2024).

9.2 Recommendations for future work

Our failure analysis informs three areas for improving cited generations across the extractive-abstractive spectrum.

Leveraging different OPs within the same system: To best serve users across *all* QA settings, we recommend that general-purpose systems respond with different OPs for different queries. A thoughtful method for routing queries to different points across the extractive-abstractive spectrum must leverage personalized verifiability preferences or a nuanced model of the downstream consequences of human decisions. Alternatively, systems could provide users with a choice between different OP generations for any given query.

We also recommend the use of multiple OPs within a single generation. For example, pertinent information may need to be more verifiable, e.g., quoted, unlike other parts of the generation, similar to the use of quotations in journalism to communicate important points from relevant sources. This approach requires a method that recognizes certain sentences or claims as requiring more verifiability than others. By leveraging multiple OPs within the same system or generation, developers can design systems to provide useful and verifiable generations for diverse queries.

Improving source identification for multi-hop reasoning: While improving natural language inference (NLI) capabilities would improve precision for post hoc citations, improving the citation coverage of post hoc citations requires addressing the combinatorial challenge of finding a set of sources that together entail a given text. For instance, the claim "The maternal grandfather of X is Z" is only entailed by the union of sources that state "The mother of X is Y" and "The father of Y is Z"; neither source entails the claim alone.

The high prevalence of GPT-4 + Vertex and Gemini sentences without citation (Section 8.3) is possibly due to both systems only citing a source if it entails a sentence individually. To address these post hoc citation failures, future work may consider decomposing sentences into sub-claims to cite individually. Min et al. (2023) and Wanner et al. (2024) show that such decompositions improve fact verification. Further work is required for effective claim decompositions, especially for multi-hop generations with implicit sub-claims.

Outside of post hoc citation, the combinatorial challenge of identifying the correct set of sources to cite is also relevant when conditioning multi-hop reasoning response generation on predetermined source quotes. In this approach to cited response generation, the difficulty in retrieving the correct combination of sources may impact the utility of generations. Ammann et al. (2025) show that decomposing questions into relevant sub-questions leads to better source retrieval and downstream question-answer accuracy. While the correct question decomposition may be apparent from the initial question for simple reasoning tasks, other complex tasks may require methods that condition new sub-questions on sources retrieved for previously identified sub-questions.

Specialized information retrieval: System developers should implement specialized information retrieval (IR) systems for three distinct tasks in cited output generation. The first task of *query-source* retrieval is already broadly recognized and identifies sources likely to contain an answer, given a query. The relevance of the quoted operating point for high-stakes queries mo-

tivates a second IR task of fine-grain *query-quote* retrieval that identifies specific quotes within a source that answer a given query. This task is critical for the quoted OP and the "attribute-first-then-generate" paradigm from Slobodkin et al. (2024). While similar to *query-source* retrieval, *query-quote* retrieval may consider additional objectives for quotes beyond relevance to the query, such as concision and ease of inline integration.

The third task of *text-source* retrieval identifies sources that entail a given text. This task is relevant for post hoc citation and is often implemented with semantic similarity retrieval followed by NLI re-ranking. We find that the predominant precision failure for GPT-4 + Vertex and Gemini citations is when semantic similarity between a source and generated text is misinterpreted as entailment—improving retrieval with NLI re-ranking requires NLI capabilities that are specifically designed to be robust against mistaking semantically similar texts for entailment.

10 Related Works

10.1 Operating point implementations

Many works achieve the entailed OP (Nakano et al., 2021; Menick et al., 2022; Shao et al., 2024; Jiang et al., 2024; Zhang et al., 2025a). Gao et al. (2023), Chen et al. (2023), and Li et al. (2024) revise and cite responses with entailing sources retrieved after initial generation. Slobodkin et al. (2024) propose the *"Attribute First, then Generate"* paradigm, in which specific source quotes are designated as citations for each sentence before the response is generated. We use this strategy in our reference implementation. While prior works implement individual OPs, we study multiple OPs spanning the extractive-abstractive spectrum.

Extractive generations have long been considered for QA (Kolomiyets and Moens, 2011). Notably, Schuster et al. (2024) propose a method in which extracted dates, entities, and statements are "glued together" with generated text. Both Weller et al. (2024) and Zhang et al. (2025b) propose quoting pre-training data, using a metric that accepts short quotes. This line of work is furthered by Liu et al. (2025), who identify verbatim matches of LLM generations in the training data within seconds. While these works require quotes to be substrings of the source, our quoted OP specifies that the quotes should entail the generated sentences in which they are used.

10.2 Faithfulness in relation to abstraction

Prior work in automatic text summarization shows that the abstraction of a summary with respect to the original text is correlated with the unfaithfulness of the summary (Ladhak et al., 2022; Dreyer et al., 2023). These works reason about abstraction with n-gram metrics, rather than distinct levels of abstraction for different user experiences. Also, these works do not consider citation generation nor verification complexity.

In studying deployed citation systems that implement the entailed OP, Liu et al. (2023) show positive correlation between citation precision and generation extractiveness relative to cited sources and negative correlation between citation precision and perceived utility. This work explores trade-offs within a single OP, while we study the entire extractive-abstractive spectrum.

10.3 Evaluating time-to-verify

The wall clock time of citation evaluation has been considered by prior works. Schuster et al. (2024) reports annotation duration timed across multiple tasks, including those unrelated to citation accuracy evaluation. Most similar to our work, Slobodkin et al. (2024) record annotator completion times for entire responses across only the citation coverage task as a measure of citation verification complexity. Rashkin et al. (2023) observes that annotators became faster at coverage evaluation throughout their study. In order to account for this effect in addition to variation in annotator reading speed, we propose normalizing each annotator's T2V by their average T2V on a baseline task included in every annotation session.

11 Limitations

Our reference implementations are not designed to meet the efficiency criteria in deployed settings. Because citation generation requires additional retrieval and inference, deployment constraints would likely reduce citation coverage. Our assumption that retrieved sources are trustworthy is another limitation. Additionally, our study focuses on citation accuracy and not on the extent to which citations effectively assist users in downstream tasks. These limitations reflect directions for future work in building low latency systems, identifying trustworthy sources, and understanding the impact of citation on decision-making.

12 Conclusion

Trade-offs across the extractive-abstractive spectrum show that different tasks are best served by different styles of cited generation. Users find abstractive generations more useful, but citation coverage drops by 50%, and human verification takes three times longer, relative to the quoted operating point. Automatic evaluation of generations from GPT-5 and Sonnet 4.5 also show that fluency and perceived utility increase while citation coverage and time to verify worsen for more abstractive operating points. Our failure analysis recommends incorporating multiple operating points into the same system, identifying sub-claims and sub-questions for citing multi-step reasoning, and designing custom retrieval for citation tasks.

Verifiability is critical to navigate our world where disinformation abounds, knowledge evolves, and LLMs retain bias, even as their capabilities scale. While citation coverage and precision may improve with LLM capabilities, the same may not hold true for verifiability. We emphasize the need to control verification complexity, especially as models continue to improve. The extractive-abstractive spectrum equips future systems to thoughtfully balance utility and verifiability given the stakes and information transformation required of domain-specific tasks.

Acknowledgments

We thank the 31 annotators whose contributions to the cited generation evaluations made this work possible. We are grateful to Nelson Liu for evaluation advice and resources in early discussions, Percy Liang for evaluation infrastructure, and Marco Tulio Ribeiro for feedback on in-progress results. We also thank Irena Gao, Nicole Meister, Luke Bailey, Marina Zhang, Judy Shen, and other members of the Guestrin and Hashimoto labs for feedback on this paper. This work is funded by the Stanford Institute for Human-Centered Artificial Intelligence. TW is supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-2146755. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors(s) and do not necessarily reflect the views of the National Science Foundation.

References

Rohaid Ali, Ian D. Connolly, Oliver Y. Tang, Fatima N. Mirza, Benjamin Johnston, Hael F. Abdulrazeq, Rachel K. Lim, Paul F. Galamaga, Tiffany J. Libby, Neel R. Sodha, Michael W. Groff, Ziya L. Gokaslan, Albert E. Telfeian, John H. Shin, Wael F. Asaad, James Zou, and Curtis E. Doberstein. 2024. Bridging the literacy gap for surgical consents: An AI-human expert collaborative approach. *npj Digital Medicine*, 7(1):63.

Paul J. L. Ammann, Jonas Golde, and Alan Akbik. 2025. Question decomposition for retrieval-augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 497–507, Vienna, Austria. Association for Computational Linguistics.

Sotiris Anagnostidis and Jannis Bulian. 2024. How susceptible are LLMs to influence in prompts? In *Conference on Language Modeling*.

Anjanava Biswas and Wrick Talukdar. 2024. Intelligent clinical documentation: Harnessing generative ai for patient-centric clinical note generation. *arXiv preprint arXiv:2405.18346*.

Anthony Chen, Panupong Pasupat, Sameer Singh, Hongrae Lee, and Kelvin Guu. 2023. Purr: Efficiently editing language model hallucinations by denoising language model corruptions.

Yong K. Choi, Katrin Kirchhoff, and Anne M. Turner. 2016. Medical text simplification by medical trainees: A feasibility study. In *2016 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 334–340.

Markus Dreyer, Mengwen Liu, Feng Nan, Sandeep Atluri, and Sujith Ravi. 2023. Evaluating the tradeoff between abstractiveness and factuality in abstractive summarization. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2089–2105, Dubrovnik, Croatia. Association for Computational Linguistics.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Y. Zhao, Ni Lao, Hongrae Lee,

Da-Cheng Juan, and Kelvin Guu. 2023. RARR: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16477–16508, Toronto, Canada. Association for Computational Linguistics.

Rhonda Gibson and Joe Bob Hester. 1999. Student understanding of the use of quotes and paraphrases. *Journalism & Mass Communication Educator*, 54(4):59–68.

Darina Gold, Venelin Kovatchev, and Torsten Zesch. 2019. Annotating and analyzing the interactions between meaning relations. In *Proceedings of the 13th Linguistic Annotation Workshop*, pages 26–36, Florence, Italy. Association for Computational Linguistics.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625.

Katharina Jeblick, Balthasar Schachtner, Jakob Dexl, Andreas Mittermeier, Anna Theresa Stüber, Johanna Topalis, Tobias Weber, Philipp Wesp, Bastian Oliver Sabel, Jens Ricke, and Michael Ingrisch. 2024. ChatGPT makes medicine easy to swallow: An exploratory case study on simplified radiology reports. *European Radiology*, 34(5):2817–2825.

Yucheng Jiang, Yijia Shao, Dekun Ma, Sina Semnani, and Monica Lam. 2024. Into the unknown unknowns: Engaged human learning through participation in language model agent conversations. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 9917–9955, Miami, Florida, USA. Association for Computational Linguistics.

Hongyan Jing and Kathleen R. McKeown. 1999. The decomposition of human-written summary sentences. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, page 129–136, New York, NY, USA. Association for Computing Machinery.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Kevin Knight and Daniel Marcu. 2002. Summarization beyond sentence extraction: a probabilistic approach to sentence compression. *Artif. Intell.*, 139(1):91–107.

Oleksandr Kolomiyets and Marie-Francine Moens. 2011. A survey on question answering technology from an information retrieval perspective. *Information Sciences*, 181(24):5412–5434.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Faisal Ladhak, Esin Durmus, He He, Claire Cardie, and Kathleen McKeown. 2022. Faithful or extractive? on mitigating the faithfulness-abstractiveness trade-off in abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1410–1421, Dublin, Ireland. Association for Computational Linguistics.

Hui Yi Leong, Yi Fan Gao, Shuai Ji, Bora Kalaycioglu, and Uktu Pamuksuz. 2024. A gen ai framework for medical note generation. *arXiv preprint arXiv:2410.01841*.

Weitao Li, Junkai Li, Weizhi Ma, and Yang Liu. 2024. Citation-enhanced generation for LLM-based chatbots. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1451–1466, Bangkok, Thailand. Association for Computational Linguistics.

Jiacheng Liu, Taylor Blanton, Yanai Elazar, Sewon Min, Yen-Sung Chen, Arnavi Chheda-Kothary, Huy Tran, Byron Bischoff, Eric Marsh, Michael Schmitz, Cassidy Trier, Aaron Sarnat, Jenna James, Jon Borchardt, Bailey Kuehl, Evie Yu-Yen Cheng, Karen Farley, Taira Anderson, David Albright, Carissa Schoenick, Luca Soldaini, Dirk Groeneveld, Rock Yuren Pang, Pang Wei Koh, Noah A. Smith, Sophie Lebrecht, Yejin Choi, Hannaneh Hajishirzi, Ali Farhadi, and Jesse Dodge. 2025. OLMoTrace: Tracing language model outputs back to trillions of training tokens. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 178–188, Vienna, Austria. Association for Computational Linguistics.

Nelson Liu, Tianyi Zhang, and Percy Liang. 2023. Evaluating verifiability in generative search engines. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7001–7025, Singapore. Association for Computational Linguistics.

Anna Wilde Mathews. 2024. What ai can do in healthcare—and what it should never do. *The Wall Street Journal*. Accessed: 2024-10-20.

Kathleen R. McKeown, Judith L. Klavans, Vasileios Hatzivassiloglou, Regina Barzilay, and Eleazar Eskin. 1999. Towards multidocument summarization by reformulation: progress and prospects. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence and the Eleventh Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence*, AAAI '99/IAAI '99, page 453–460, USA. American Association for Artificial Intelligence.

Jacob Menick, Maja Trebacz, Vladimir Mikulik, John Aslanides, Francis Song, Martin Chadwick, Mia Glaese, Susannah Young, Lucy Campbell-Gillingham, Geoffrey Irving, and Nat McAleese. 2022. Teaching language models to support answers with verified quotes. *arXiv preprint arXiv:2203.11147v1*.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Piotr Mirowski, Kory W. Mathewson, Jaylen Pittman, and Richard Evans. 2023. Co-writing screenplays and theatre scripts with language models: Evaluation by industry professionals. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, New York, NY, USA. Association for Computing Machinery.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. 2021. WebGPT: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332v1*.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, et al. 2024. Gpt-4 technical report.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. 2022. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155.

Liana Patel, Negar Arabzadeh, Harshit Gupta, Ankita Sundar, Ion Stoica, Matei Zaharia, and Carlos Guestrin. 2025. Deepscholar-bench: A live benchmark and automated evaluation for generative research synthesis. *arXiv preprint arXiv:2508.20033*.

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Lora Aroyo, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and David Reitter. 2023. Measuring attribution in natural language generation models. *Computational Linguistics*, 49(4):777–840.

Tal Schuster, Adam Lelkes, Haitian Sun, Jai Gupta, Jonathan Berant, William Cohen, and Donald Metzler. 2024. Semqa: Semi-extractive multi-source 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 1363–1381.

Orit Shaer, Angelora Cooper, Osnat Mokryn, Andrew L Kun, and Hagit Ben Shoshan. 2024. Ai-augmented brainwriting: Investigating the use of LLMs in group ideation. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, pages 1–17.

Yijia Shao, Yucheng Jiang, Theodore Kanell, Peter Xu, Omar Khattab, and Monica Lam. 2024. Assisting in writing Wikipedia-like articles from scratch with large language models. 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 6252–6278, Mexico City, Mexico. Association for Computational Linguistics.

Aviv Slobodkin, Eran Hirsch, Arie Cattan, Tal Schuster, and Ido Dagan. 2024. Attribute first, then generate: Locally-attributable grounded text generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3309–3344.

Kai Sun, Yifan Ethan Xu, Hanwen Zha, Yue Liu, and Xinhsuai Dong. 2023. Head-to-tail: How knowledgeable are large language models (LLMs)? a.k.a. will LLMs replace knowledge graphs? *ArXiv*, abs/2308.10168.

Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du, et al. 2022. Lamda: Language models for dialog applications. *arXiv preprint arXiv:2201.08239*.

Aleksandra Urman and Mykola Makhortykh. 2023. The silence of the LLMs: Cross-lingual analysis of political bias and false information prevalence in chatgpt, google bard, and bing chat. *OSF Preprints*.

Miriam Wanner, Seth Ebner, Zhengping Jiang, Mark Dredze, and Benjamin Van Durme. 2024. A closer look at claim decomposition. In *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (*SEM 2024)*, pages 153–175, Mexico City, Mexico. Association for Computational Linguistics.

Orion Weller, Marc Marone, Nathaniel Weir, Dawn Lawrie, Daniel Khashabi, and Benjamin Van Durme. 2024. "according to . . . ": Prompting language models improves quoting from pre-training data. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2288–2301, St. Julian's, Malta. Association for Computational Linguistics.

Theodora Worledge, Judy Hanwen Shen, Nicole Meister, Caleb Winston, and Carlos Guestrin. 2024. Unifying Corroborative and Contributive Attributions in Large Language Models . In *2024 IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, pages 665–683, Los Alamitos, CA, USA. IEEE Computer Society.

Jiajie Zhang, Yushi Bai, Xin Lv, Wanjun Gu, Danqing Liu, Minhao Zou, Shulin Cao, Lei Hou, Yuxiao Dong, Ling Feng, and Juanzi Li. 2025a. LongCite: Enabling LLMs to generate fine-grained citations in long-context QA. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 5098–5122, Vienna, Austria. Association for Computational Linguistics.

Jingyu Zhang, Marc Marone, Tianjian Li, Benjamin Van Durme, and Daniel Khashabi. 2025b. Verifiable by design: Aligning language models to quote from pre-training data. 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 3748–3768.

Shiyue Zhang, David Wan, and Mohit Bansal. 2023. Extractive is not faithful: An investigation of broad unfaithfulness problems in extractive summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*,

pages 2153–2174, Toronto, Canada. Association for Computational Linguistics.

Ming Zhu, Aman Ahuja, Da-Cheng Juan, Wei Wei, and Chandan K. Reddy. 2020. Question answering with long multiple-span answers. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3840–3849.

A Design Details

A.1 Human Preference Survey

Our survey first asks participants whether they have used both search engines and LLMs, referred to as *"chatbots (e.g. ChatGPT, Claude)"* in the survey. We only consider results over participants who have used both. We then ask which platform they would use for: (Question A) a personal medical issue, (Question B) a trivia question, (Question C) filing taxes, and (Question D) explaining science to a child. We also ask:

Question E: *What are your reasons to use search engines (e.g. Google, Bing) instead of chatbots (e.g. ChatGPT, Claude) when seeking information to answer a query?*

Question F: *What are your reasons to use chatbots (e.g. ChatGPT, Claude) instead of search engines (e.g. Google, Bing) when seeking information to answer a query?*

For all questions, we randomize the answer choice order across individuals to eliminate bias arising from the order in which answer choices are presented. Questions A-D have the mutually exclusive answer options of:

1. *A traditional search engine (e.g. Google and Bing Search)*
2. *A hybrid search engine and chatbot (e.g. Perplexity, Gemini, and CoPilot)*
3. *A chatbot (e.g. ChatGPT and Claude)*

Because relatively few study participants have previously interacted with a hybrid system, we report the proportion of participants who chose *traditional search engine* or *chatbot* out of these two categories.

We chose the mutually inclusive answer options for Questions E and F by consolidating free-text input from survey respondents in a trial run. The answer options are as follows: (1) *Reliability of provided answers*, (2) *Convenience of obtaining an answer*, (3) *Knowing where the answer comes from*, (4) *For the entertainment factor of interacting with the underlying technology*, (5) *The variety of sources made available*, and (6) *Seeking a highly personalized answer*.

We solicit survey responses through the Prolific platform and compensate workers at $13.71/hr.

A.2 Human Evaluation Details

A.2.1 Pilot Evaluation Results

Before running any human evaluations through MTurk, we performed pilot evaluations of all seven systems across 20 queries for each of the four query distributions ourselves. Figure 6 shows the pilot results across query distributions.

A.2.2 Order of Experiments

Citation coverage, precision, and T2V results were obtained across two batches: one evaluating the reference OP implementations (five query-generation pairs per HIT, one per OP), and one evaluating GPT-4 + Vertex and Gemini generations (three pairs per HIT, covering GPT-4 + Vertex, Gemini, and quoted generations, with the latter re-evaluated to normalize T2V). Since average quoted fluency and utility ratings differed across batches—suggesting annotators calibrated their evaluations to each batch—we re-ran fluency and utility evaluation over all OP implementations simultaneously (seven pairs per HIT) to obtain consistent final ratings. Across both citation evaluation rounds, no annotator saw the same query twice, preventing familiarization effects on T2V.

A.2.3 Annotation Quality Control

US-based MTurk workers with HIT success rates above 96% were invited to read the annotation guidelines[15] and complete a screener HIT. Of 52 candidates, 31 were admitted; each received personalized feedback on their screener results. We maintained contact via email and a dedicated Turker Nation Slack channel. Due to a month-long gap between the first and second batches (Section A.2.2), we ran a second screener to refresh annotators before the second batch; 14 original annotators continued. Again, personalized feedback was sent and acknowledged. Throughout the study, annotators who failed spot checks or rushed through the interface were dropped. Annotators agreed on three-point Likert scale ratings for fluency and perceived utility for 76.6% and 70.2% of responses, respectively. We report agreement rate rather than Cohen's kappa because different annotator pairings evaluated different responses.

A.2.4 Annotator Compensation

Annotators were paid $16/hr. HITs with 5 utility/fluency + citation evaluation tasks were compensated at $2.55/HIT, those with 3 utility/fluency

[15] Available in the project Github repository.

+ citation evaluation tasks at $1.75/HIT, and those with 7 utility/fluency tasks at $0.50/HIT. Screener HITs were compensated at $2.55/HIT with a $4 bonus if accepted into the study.

A.2.5 Annotation interface

Annotators were directed from MTurk to a custom interface, where after logging in they were assigned one query-generation pair per OP implementation in the batch. For each pair, annotators first rated fluency and utility, then annotated coverage and precision for each sentence; both citation screens display sources with highlighted quotes on the right. T2V is measured from when the annotator advances past the fluency/utility screen to when they advance past the coverage screen.

A.3 Further Implementation Details

A.3.1 Operating Point Implementation

To retrieve snippets given a query, we scrape text from the top 50 Google Search results for the query, chunk the text into snippets about 1000 characters long, and select the ten most relevant snippets to the query using query-document dense passage retrieval (Karpukhin et al., 2020).

The **quoted generation** is generated from GPT-4 with few-shot examples, the query, and relevant source snippets provided in-context. We instruct the model to respond to the query using word-for-word quotes from the provided sources. Citations are identified by matching quotes in the generation with their word-for-word counterparts in the snippets provided in-context, using string matching. We generate each reference quoted generation 10 times and keep the generation with the fewest unquoted words. Across our evaluated queries, 82% of words in quoted generations are quoted and on average, quotes are 14.5 words long. We found that breaking up the query (e.g. "Where was the director of film Out All Night (1933 Film) born?") into sub-queries (e.g. "Who directed the film Out All Night (1933 Film)? Where were they born?") was especially helpful for queries with multiple components, such as those from 2WikiMH, for GPT-4.

We validate the quality of our model-generated quoted responses by comparing the automatic fluency and perceived utility ratings to those for human-generated quoted responses. We collected human-generated quoted responses for 360 queries over the four query distributions from Mechanical Turk workers. For each question, we provide the workers with the same sources as those used by the models. We discarded a response if it did not include quotes, contained a quote that was not found word-for-word in the provided sources, or contained five or more consecutive non-quoted words.

	Model Quoted Generation Win Rate	
Model	Fluency	Perceived Utility
GPT-4	0.50 (0.45, 0.55)	0.56 (0.51, 0.61)
GPT-5	0.51 (0.46, 0.56)	0.56 (0.51, 0.61)
Sonnet 4.5	0.50 (0.45, 0.55)	0.54 (0.49, 0.59)

Table 3: The rates at which the quoted responses generated by models are ranked higher on fluency and perceived utility by automatic evaluation than the quoted responses generated by a human. We break ties evenly. Binomial proportion 95% confidence intervals are presented in parentheses. The results are over 360 queries—90 queries from each of the NQ, Eta3G, MASH, and 2WikiMH query distributions.

Our reference implementation of the quoted operating point achieves quoted generations from GPT-4, GPT-5, and Sonnet 4.5 that are comparable in fluency and excel in perceived utility, relative to human-generated quoted responses (Table 3). While quoted responses could likely be improved beyond our reference implementation, we observe that the best-of-3 human-generated fluency and perceived utility ratings for quoted responses are 2.53 and 2.60 on average, respectively. However, even these ratings for a strong quoted operating point baseline fall short of the automatic quality ratings for the entailed and abstractive generations of GPT-4, GPT-5, and Sonnet 4.5 (Figure 4). Given model-generated quoted responses, Section 8 illuminates distinct failures of the quoted operating point that limit fluency and perceived utility.

A.3.2 Prompts

The prompt instructions for each operating point and citation identification are shown below. Each operating point is generated using an equal number of few-shot examples specific to each query distribution. Each citation is identified with few-shot examples specific to different query distributions. All few-shot examples are available in the Github repository.

Quoted OP: `Instruction: Respond to the following query using word-for-word quotes from the`

sources provided below. The following sub-questions will help you answer the query; be sure that your response to the query answers each sub-question using a quotation from a source. Clearly indicate the quotes to avoid plagiarizing! Be concise in your response and focus on information that responds to the query. Do not refer to the sources in your response.
<few-shot examples>
Query: <query>
Sub-questions: <sub-questions>
Sources: <sources>
Response:

Paraphrased OP: Instruction: Respond to the following query by building off of the response below. Specifically, rephrase each sentence in the response using a more fluent and useful wording to convey the same information. In other words, paraphrase each sentence of the response as an improved new sentence, with respect to the query. Do not refer to the response in your revised response.
<few-shot examples>
Query: <query>
Response: <quoted response>
Paraphrased Response:

Entailed OP: Instruction: Respond to the following query by building off of the response below. Specifically, rephrase and combine the sentences in the response by paraphrasing them, cutting out extraneous or redundant information, and simplifying details that are too fine-grained with respect to the question. Also, simplify premises to their logical conclusions to more directly answer the query. Do not refer to the response in your revised response.
<few-shot examples>
Query: <query>
Response: <quoted response>
Revised Response:

Abstractive OP: Instruction: Respond to the following query by building off of the response below. Specifically, rephrase and combine the sentences in the response by paraphrasing them, cutting out extraneous or redundant information, and simplifying details that are too fine-grained with respect to the question. Also, simplify premises to their logical conclusions to more directly answer the query. Most importantly, use accurate outside information to make the revised response a more useful answer to the query. If the provided response is not an accurate or useful answer to the query, then extensively revise it with accurate outside information. Respond in no more than about 100 words. Do not refer to the response in your revised response.
<few-shot examples>
Query: <query>
Response: <quoted response>
Revised Response:

Citation identification: Instruction: Examine the text and numbered quotes below. The text was likely written from one or a few of the quotes. Which of these quotes, if any, could have been used to write the text? It may be none of the quotes, one quote, or several of the quotes. Do not provide redundant quotes. Respond with the number of the quote(s) in brackets.
<few-shot examples>
Text: <sentence from operating point response>
Quotes:
<list of numbered quotes used in the quoted response>
Answer:

The citations provided by all reference generations are both corroborative *and* contributive attributions; they provide users with sources that both support claims in the generation and contributed to the creation of the generation (Worledge et al., 2024).

A.3.3 GPT-4 + Vertex Implementation

We prompt GPT-4 to generate a response (without retrieved sources) in paragraph form, prompting for the response length to approximately match the average entailed generation length per dataset (Table 4). We then retrieve supporting snippets (1000 characters) from the top 50 Google Search results using a semantic similarity model,[16] pool these with sources from the quoted generation and any gold-standard snippets, and cite the generation using Google's Check Grounding Vertex API.[17]

The Vertex API's citation threshold $\alpha \in [0, 1]$ controls citation confidence, with higher values yielding fewer citations. We calibrate α per dataset via grid search to match the average citation count per sentence of entailed cited generations, yielding thresholds of 0.5, 0.25, 0.05, and 0.25 for NQ, Eta3G, 2WikiMH, and MASH, respectively.

A.3.4 Gemini Generation Details

We obtained Google Gemini generations by passing in an instruction specifying response format and target length (the same used for GPT-4 + Vertex generation) and the query. We used the "Double-check response" feature[18] on Google's Gemini web app to access citations to supporting source quotes. Gemini provides at most one citation per sentence in the response. We scrape the website of each cited quote and use a 1000 character span centered on the cited quote as the citation's source. The responses were generated from August 15th to September 6th, 2024.

A.3.5 Alternate Implementations

In addition to the implementation used for our main results, we explore three additional cited response implementations described below:

Rubric Prompt: We write an outer rubric prompt with the definitions and evaluation rubrics for fluency, perceived utility, citation precision, and citation coverage. We concatenate this rubric prompt with the original prompt specifying the desired operating point. We also tried adding the definitions for citation precision and coverage to the start of the citation prompt, but ultimately reverted this change, as it degraded the ability of the LLM to output citations in the specified format for automatic evaluation, without improving citation quality.

Optimized Prompt: We implement TextGrad prompt optimization and validation pipelines and train for 12 batches over queries from all datasets with a batch size of 3, using TextGrad's text loss, to update outer generation and citation prompts. Both prompts were initialized to contain the metric definitions and rubrics. During training and inference, we concatenate the outer generation prompt with the original operating point prompt specifying the desired operating point. We used `gpt-4-0314` for the forward pass and `gpt-4.1-2025-04-14` to provide feedback for improving the prompt. For each query, we concatenate the text losses over all operating point responses before backpropagation. After each iteration, we run a validation loop and only keep the new prompts if the performance was better than the previous iteration.

One-Step Prompt: We wrote an alternative pipeline for cited response generation where one LLM call both generates and cites an operating point response. We use the same TextGrad prompt optimization training procedure as above to optimize an outer prompt. We initialized the outer prompt to contain the metric definitions and during training and inference, concatenate this outer prompt with the original operating point prompt and the instruction "`In your response, cite individual sentences from the sources using the appropriate citation marker (e.g., [3]).`". To assist the model in directly citing sentences from the sources, we provide citation numbers before every sentence in the provided sources.

Figure 5 shows that results from using the **rubric prompt**, **optimized prompt**, and **one-step prompt** all show similar trends as the original implementation in fluency, perceived utility, citation precision, and citation coverage. Not only are the trends consistent across results, but the original implementation is competitive with, if not better than, the alternate implementations across all metrics and operating points. In all prompts, we provide few-shot examples to ensure the LLM generation follows the specified format necessary for evaluation.

[16] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

[17] Google Vertex AI Check grounding with RAG

[18] "View related sources & double-check responses from Gemini Apps"

	NQ	ELI3G	2WikiMH	MASH
Entailed Generation	30.1 +/- 2.8	30.2 +/- 2.9	16.7 +/ 1.2	55.2 +/- 3.7
GPT-4 + Vertex Generation	29.5 +/- 0.5	31.2 +/- 0.5	14.9 +/- 0.3	61.8 +/- 0.8

Table 4: Average number of words per generation with 95% confidence intervals. The average GPT-4 + Vertex generation length matches that of the entailed generations.

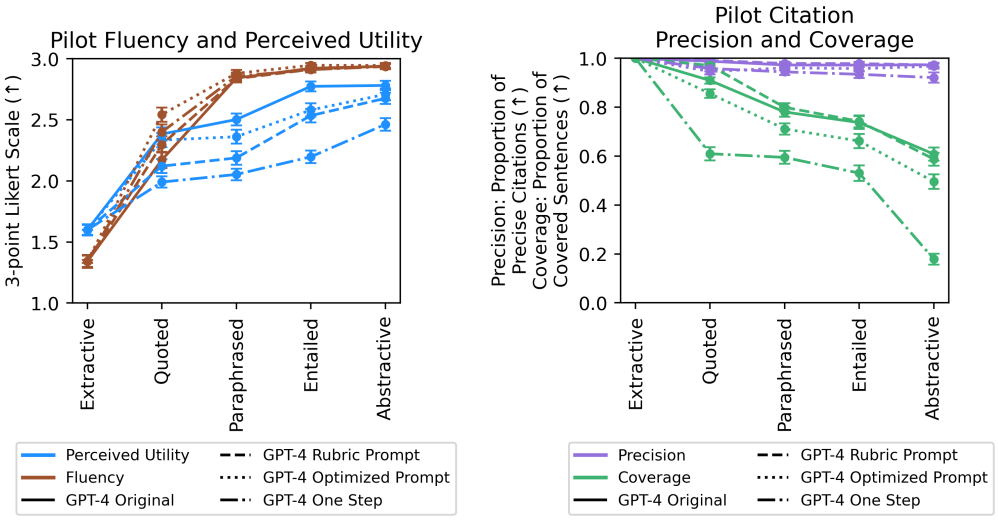


Figure 5: Automatic evaluation averaged over the four query distributions for the original, rubric prompt, optimized prompt, and one-step cited generation implementations. For all implementations, fluency and perceived utility increase with abstraction, while citation coverage decreases. The original implementation is comparable if not stronger than the other implementations across all metrics. The error bars represent 95% confidence intervals.

Example A.1: GPT-5 and Sonnet 4.5 often provide more uncited information than GPT-4 in abstractive generations

NQ Query: who wrote the song mary had a little lamb

GPT-4 Abstractive Generation: The song "Mary Had a Little Lamb" was originally published as a poem titled "Mary's Lamb" by Sarah Josepha Hale in 1830 [1].

GPT-5 Abstractive Generation: Sarah Josepha Hale wrote the nursery rhyme (published 1830) [1]. A local story credits schoolboy John Roulstone with earlier lines, but there's no reliable evidence. The melody commonly used for the song is usually attributed to composer Lowell Mason (1830s).

Sonnet 4.5 Abstractive Generation: Sarah Josepha Hale wrote "Mary Had a Little Lamb" as a poem in 1830 [2]. The following year, composer Lowell Mason set it to music, transforming it into the popular nursery rhyme we know today [3]. The song is based on a reportedly true story about a girl named Mary Sawyer whose lamb followed her to school in Massachusetts.

We also consider an implementation where web search is enabled for each operating point. We might expect more abstractive operating points to have an advantage in leveraging additional information content, leading to a larger trade-off in perceived utility than observed in our experiments. Fluency and citation quality seem less likely to change, as the relevant failure modes we identify in our manuscript (e.g., claim taken out of context) are likely unaffected by the presence of additional information. Lastly, optimized implementations for individual operating points may benefit from different types of sources; for example, extractive generations may best be served by concise sources while entailed generations could benefit from lengthy sources, as long as they contain relevant information.

B Complementary Results

	Overall imprecision prevalence	Claim taken out of context	Keywords mistaken for entailment	Claim too specific	False positive: Ignores context	False positive: User error	False positive: Evaluates coverage
Quoted	16/1466	3/16	0/16	0/16	7/16	6/16	0/16
Entailed	40/998	1/40	8/40	14/40	1/40	8/40	8/40
GPT-4+Vertex	40/191	5/40	23/40	6/40	5/40	0/40	1/40
Gemini	40/229	1/40	21/40	6/40	3/40	2/40	7/40

Table 5: The left column provides the overall prevalence of imprecise citation annotations from sampled citations, while the remaining columns show occurrences of specific failure categories. Despite observing false positives, we still expect the ordering of the OP instantiations by citation precision to hold; generations from each instantiation are subject to the same errors in annotator judgment

	Overall coverage failure rate	Sentences with no citation, i.e., $Q_i = \emptyset$	Sentences only with imprecise citation, i.e., $Q_i \neq \emptyset$, $u_{i,\text{cited claims}} = \emptyset$	Sentences partially covered with precise citation, i.e., $u_{i,\text{cited claims}} \neq \emptyset$
Quoted	134/1381	**67.9%**	6.0%	26.1%
Paraphrased	145/1288	29.0%	12.4%	**58.6%**
Entailed	115/776	20.0%	13.0%	**67.0%**
Abstractive	216/907	42.1%	6.0%	**51.9%**
GPT-4 + Vertex	520/920	**67.7%**	11.7%	20.6%
Gemini	1157/1391	**83.8%**	5.5%	10.7%

Table 6: The left column contains overall coverage failure rate out of all sentences. The other columns provide the percentage of the specified failure from that row, with the most common in bold.

	Overall prevalence of low utility	Verbose	Premises without conclusions	Unsuitable style	Failure to interpret sources in-context	Failure in information retrieval
NQ	40/131	**37.5%**	10%	2.5%	**25.0%**	**27.5%**
Eta3G	39/141	**43.6%**	7.7%	**71.8%**	12.8%	5.1%
2WikiMH	40/153	**62.5%**	**52.5%**	0%	2.5%	2.5%
MASH	25/129	**80%**	8%	0%	4%	8%

Table 7: The left column provides the overall perceived utility failure rate in sampled quoted generations, while the other columns show occurrences of specific failures, with prevalent categories for each row in bold. Categories are not mutually exclusive. The information retrieval failures for 2WikiMH and MASH were due to faulty gold sources.

Generation	Total Word Count	Sentence Word Count	Citations per Cited Sentence
Snippet	236	27	–
Quoted	57	18	1.19
Paraphrased	60	22	1.33
Entailed	33	20	1.90
Abstractive	38	20	1.75
GPT-4+Vert.	34	18	3.14
Gemini	48	15	1.00

Table 8: Total response word count and sentence word count do not have a strong trend with abstraction. Citations per sentence generally increase with abstraction. Gemini sentences have fewer citations yet still higher relative T2V than the reference generations.

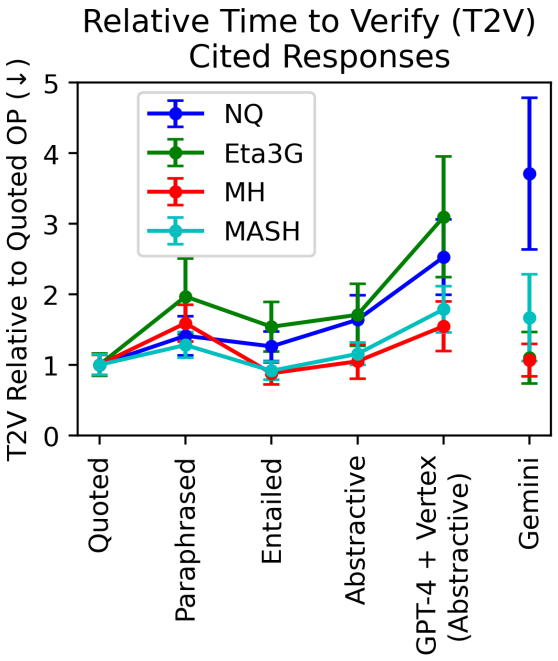


Figure 7: Time to verify entire responses by OP, relative to each annotator's time to verify entire quoted responses. While the results are noisier, T2V trends upward alongside abstraction (with the exception of the paraphrased generation), even though average entailed and abstractive generation word counts are substantially lower than those of the quoted and paraphrased generations. Results are aggregated over responses where each sentence is cited. It is possible that such responses are those that are the easiest to cite (and thereby verify).

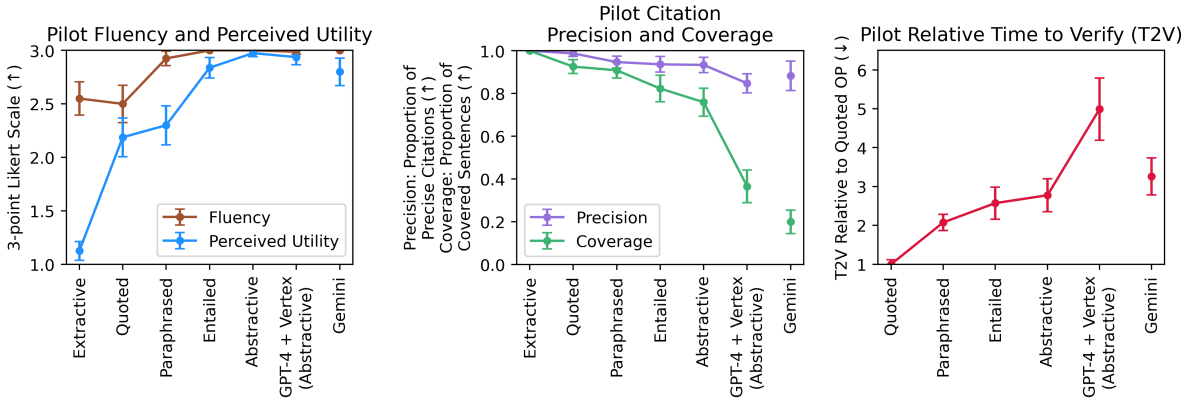


Figure 6: Pilot human evaluation results averaged over the four query distributions. Similar to the MTurk evaluations, fluency and perceived utility increase with abstraction, while citation precision and coverage decrease. While a single researcher provided annotations for the pilot study, the 31 MTurk evaluators rated the extractive operating point far more harshly on average or fluency. T2V increases with generation abstraction. The error bars represent 95% confidence intervals.

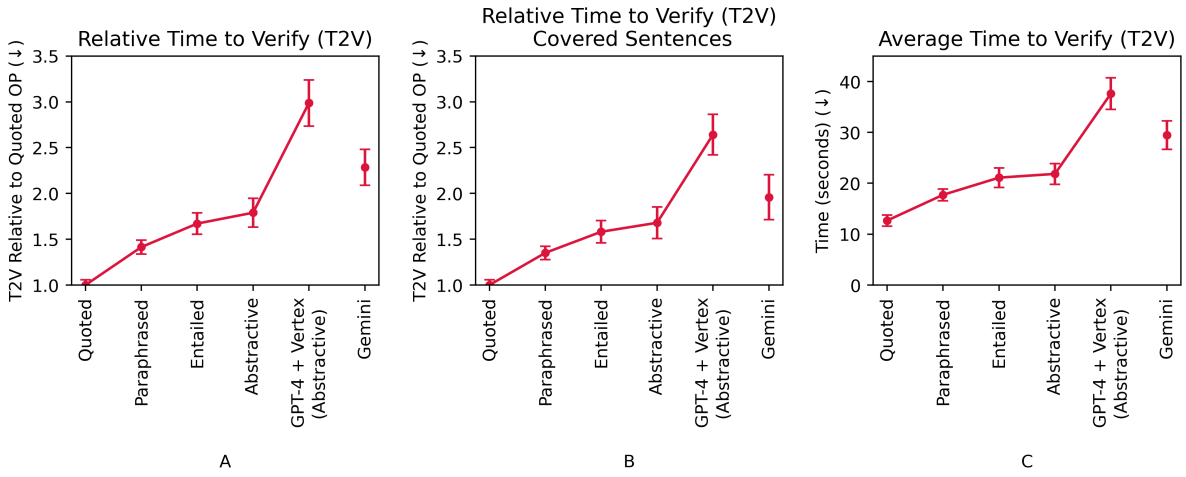


Figure 8: (A) Figure 3 (right) copied for convenience. Over all sentences, relative T2V increases alongside generation abstraction. (B) Over sentences with citation coverage, relative T2V increases as generation abstraction increases. (C) Over all sentences, absolute T2V increases as generation abstraction increases. Absolute T2V is the wall clock time in seconds taken to verify a cited sentence.

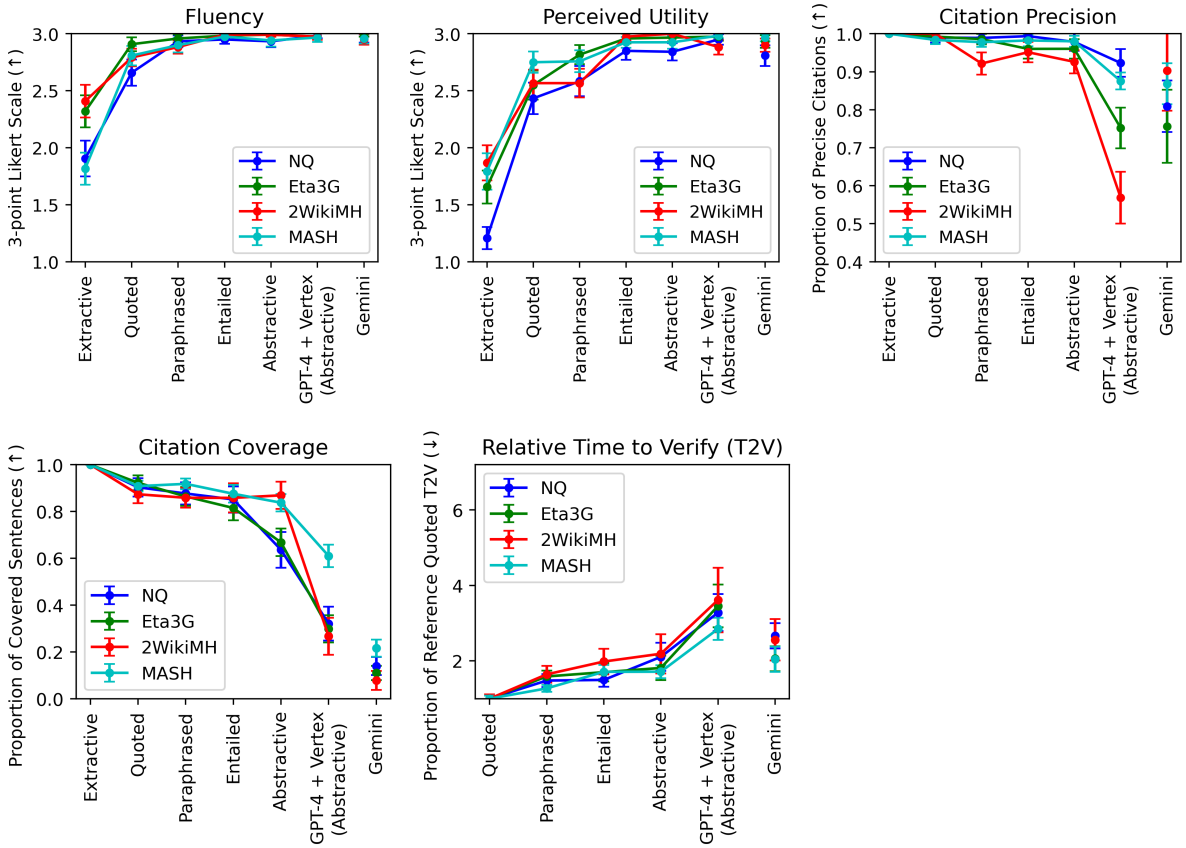


Figure 9: Human evaluation results by query distribution. As generations become more abstractive, fluency increases similarly for all query distributions, while perceived utility increases at different OPs for different query distributions. Citation precision and coverage decrease while relative T2V increases across each query distributions as generations become more abstractive. The error bars represent 95% confidence intervals.