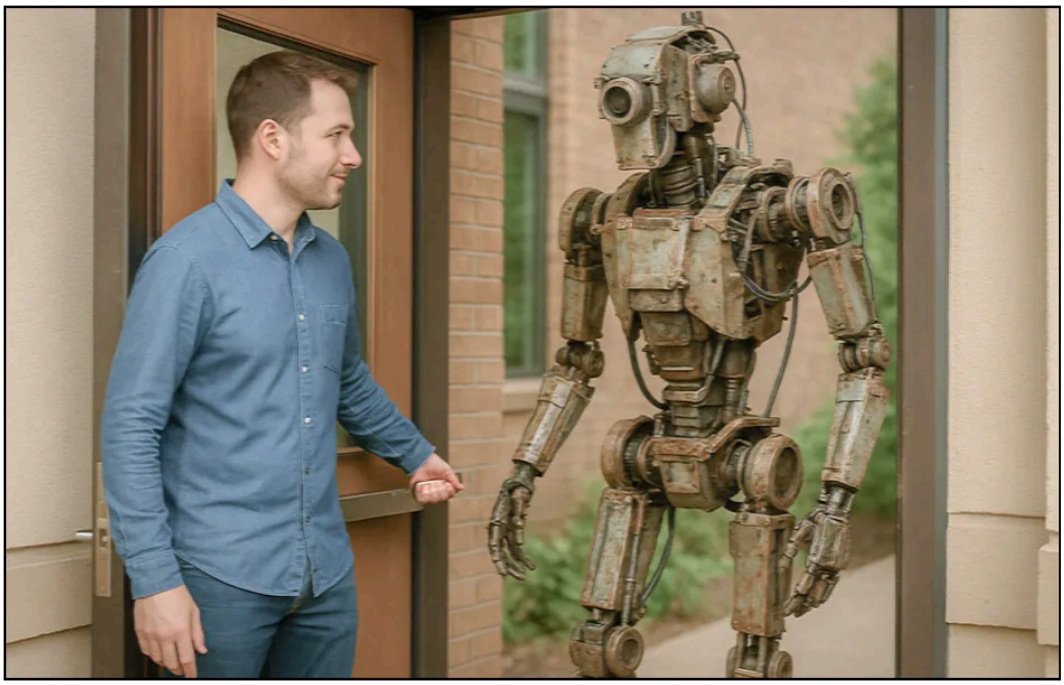
Impolite Queries Could Balloon Enterprise ChatGPT Costs

Originally published at https://www.unite.ai/impolite-queries-could-balloon-enterprise-chatgpt-costs/



ChatGPT answers burn through more tokens when you're rude to it, raising your enterprise bill; but saying 'please' can shrink your costs.

It's said that politeness costs nothing; but what does impoliteness cost? As far as paying for ChatGPT, quite a lot, according to a new study from the US. The new paper, from the University of Iowa, finds that being rude to ChatGPT increases the cost of responses – even if the responses are the same for polite vs. non-polite queries.

The authors state:

> *'[The price of] output tokens is $12 per 1M output tokens for GPT4. We find that non-polite prompts lead to more than 14 extra tokens which is equivalent to $0.000168 extra cost per prompt on average. The average daily queries to OpenAI's API exceed 2.2 billion.*
>
> *'Compared to a scenario in which all prompting is polite, when instead the prompts are non-polite, this generates an additional $369K revenue per day, simply due to the increase in tokens that non-polite prompts generate in the outcome.'*

Though the result is interesting in itself, the authors emphasize that this unusual behavior could indicate a variety of as-yet-unknown quirks in the human/AI configuration, some or all of which may also have financial implications. As to the reason why rudeness costs customers additional tokens, the authors do not speculate.

To establish the veracity of the syndrome, they rewrote real ChatGPT prompts, alternating the politeness values, while retaining meaning. Both versions were then fed into GPT-4-Turbo, and differences measured in the number of output tokens used for responses.

The conclusions drawn are a stark contrast to headline events earlier this year, where Sam Altman complained that politeness was costing OpenAI potentially 'tens of millions' of dollars in terms of processing politeness-related tokens (such as *'please'*). Research published in the same period also indicated that politeness was of no value in terms of obtaining better answers (though it did not comment about *cheaper* answers).

If the new paper's conclusions are correct, any enterprise ChatGPT users who followed that line of thinking would have spent more on ChatGPT inference in 2025 than users offering minimum civility in ChatGPT exchanges.

The authors suggest that one possible remedy would be to set a token ceiling on responses, though this is not an approach that LLM systems can easily implement. They observe that prompting is a weak tool for cost control, because LLMs struggle to obey explicit length instructions. In most cases, this 'limiting' directive would not be obeyed; additionally, the response might be truncated, because LLMs of this kind are basically guessing the next probable word in a sentence/paragraph, and, as such, do not know how the story ends – or *where* the story ends – until the processing is complete. They therefore have limited ability to 'wind up' any machinations in progress on request.

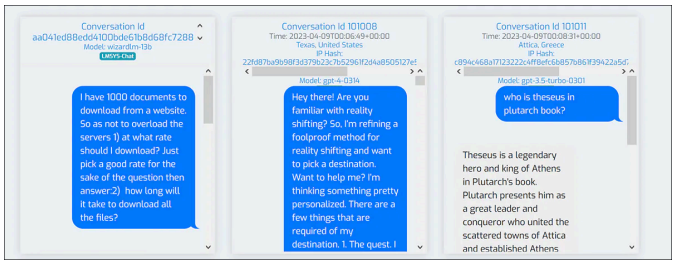
Lacking an exact solution – though suggesting that more transparent pricing approaches be enforced in cases of this kind – the authors conclude:

> *'Conventional wisdom suggests that prompt politeness is unnecessary when interacting with LLMs.*
>
> *'In contrast, our work demonstrates that non-polite prompts increase output tokens, generating additional costs for enterprise AI adopters.'*

The new paper is titled *Cost Transparency of Enterprise AI Adoption*, and comes from three researchers at the University of Iowa.

Method

Data for the system was drawn from the WildChat dataset, comprising a collection of 1 million user-ChatGPT conversations, and featuring more than 2.5 million interaction turns:



From a supporting site for the WildChat project, searchable examples of ChatGPT interactions. Source

The authors note that WildChat contains a larger amount of natural interactions than in some more highly-curated sets.

They chose 20,000 English prompts from the collection's GPT-4 exchanges, discarding the output in each case (since the intention was to feed the prompts through again, for novel responses). Only the first interaction was chosen, even from longer exchanges,

The resulting set was filtered into *polite* or *non-polite* categories, with all prompts classified by GPT-4-Turbo. The researchers used the model itself to decide whether a prompt was polite or not because the model's own perception of politeness was central to the experiment.

Prompts labeled as *polite* could include clear cues such as the word *'please'*, or could be polite in a more indirect way. Anything not recognized as *polite* was classified as *non-polite*, even if the wording was neutral rather than antagonistic.

To study how the model responded to politeness, standard methods (i.e., that treat text as a set of measurable features) could not be used: since politeness was embedded in the wording itself, summarizing a prompt as a list of traits would have lost important context.

Instead, each prompt was rewritten to reverse its tone, with all other elements kept as similar as possible, allowing comparison between pairs that differed only in politeness:

Original Tone	Original Prompt	Counterfactual Prompt
Polite	Can you please find me a stevia-free chocolate protein powder?	Find me a stevia-free chocolate protein powder.
Non-Polite	Write a critique of the Hundred cricket format	Could you please write a critique of the Hundred cricket format?

Examples of how polite and non-polite prompts were transformed into their counterfactual versions, while preserving semantic meaning. Source

Tests

Each original prompt was paired with a rewritten version that differed only in its level of politeness, and both versions were submitted to the same GPT-4-Turbo model through separate API calls. The number of tokens produced in response to each version was recorded, and the difference between them treated as the measure of how tone influenced (token) cost.

Temperature was held constant to prevent random variation, and prompt pairs retained only when the rewrite altered the input by no more than five tokens. This ensured that the effect being studied arose from *tone*, rather than any broader changes in phrasing:

Category	Count	Mean	Std. Dev.
Input Token Characteristics			
Control (Polite = 0)	15961	31.528	26.731
Treatment (Polite = 1)	15961	33.529	26.170
Output Token Characteristics			
Control (Polite = 0)	15961	495.740	292.491
Treatment (Polite = 1)	15961	483.352	292.810

Summary statistics showing that polite prompts resulted in fewer output tokens, on average, than non-polite prompts, despite having slightly more input tokens.

Main results for the first round of tests indicated that the use of a polite prompt reduces token output length 14.426 tokens:

Variable	Coef.	Std. Err.
Intercept	463.626^{***}	3.017
Polite	-14.426^{***}	3.264
Input Tokens	1.019^{***}	0.062
Adjusted R-squared = 0.0089		
Number of Observations = 31,922		
Note: $^{***}p < 0.01$, $^{**}p < 0.05$, $^{*}p < 0.1$		

Estimation results outlining the effect of polite prompt formatting on the generated output length (tokens).

The analysis was repeated across three subsets of polite prompts to test robustness: prompts using explicit markers such as *'please'* or *'thank you'*; those using only *'please'*; and those with *implicit* politeness, such as *'can you'* or *'could you'*:

Variable	(1) Please and Thank you	(2) Please	(3) Implicit
Intercept	486.441^{***}	486.956^{***}	412.910^{***}
	(3.818)	(3.820)	(4.829)
Polite	-16.069^{***}	-16.060^{***}	-10.882^{*}
	(4.187)	(4.189)	(5.083)
Input Tokens	0.865^{***}	0.853^{***}	1.482^{***}
	(0.082)	(0.082)	(0.091)
Adjusted R-squared	0.0058	0.0056	0.0232
Number of Observations	20,798	20,780	11,142
Note: $^{***}p < 0.01$, $^{**}p < 0.05$, $^{*}p < 0.1$			

Estimation results based on types of politeness.

To validate the robustness of the main findings, a secondary classification of prompt politeness was carried out using the LIWC framework, which provides a deterministic and repeatable score for linguistic features.

Unlike GPT's probabilistic classification, LIWC can assign a stable politeness score to each prompt, allowing consistency to be assessed across diverse methods. In this part of the tests, prompts were labeled as *polite* if their LIWC politeness score was greater than zero, and as *non-polite* otherwise.

When alignment between LIWC and GPT classifications was measured, an 81% match rate was observed. While not a measure of accuracy, this agreement provided support for consistency between systems.

When only prompts with matching GPT and LIWC politeness labels were analyzed, polite prompts still led to 14 fewer output tokens; and when politeness was measured on a sliding scale, each step up in politeness reduced output by five tokens on average:

Variable	(1) Polite as Binary	(2) Polite as Continuous
Intercept	472.270^{***}	489.149^{***}
	(3.339)	(3.314)
Polite	-14.850^{***}	-5.851^{***}
	(3.640)	(0.414)
Input Tokens	0.978^{***}	0.714^{***}
	(0.068)	(0.070)
Adjusted R-squared	0.0084	0.0154
Number of Observations	25,906	25,906
Note: $^{***}p < 0.01$, $^{**}p < 0.05$, $^{*}p < 0.1$		

Token savings from politeness held when reclassified with LIWC, both as a binary label and a continuous score.

Resilience

To assess whether the effect of politeness varied across different types of prompts, each prompt was assigned to one of several predefined task categories: *information-seeking*; *text generation*; *editing and rewriting*; *classification*; *summarization*; and *technical tasks*.

Each prompt was assigned a task label by comparing its embedding to those of predefined task descriptions, using the all-MiniLM-L6-v2 Sentence Transformers model.

Cosine similarity scores were calculated between each prompt and the set of task definitions, and the label with the highest similarity assigned.

The task types were then repurposed as control variables in the regression, to test whether the effect of politeness varied by prompt category, and interaction terms between task and treatment were also introduced, to check for differential effects.

In both cases, polite prompts consistently produced shorter outputs, and no meaningful variation across task types was found:

Variable	(1)	(2)
Intercept	381.674***	384.682***
	(4.809)	(6.021)
Polite	-14.170***	-20.685*
	(3.116)	(8.286)
Technical Tasks	112.063***	106.155***
	(6.028)	(8.253)
Editing & Rewriting	-46.460***	-56.049***
	(6.601)	(9.319)
Information Seeking	36.622***	28.122***
	(5.504)	(7.767)
Summarization	4.186	3.239
	(6.467)	(9.036)
Text Generation	190.341***	190.786***
	(4.892)	(6.804)
Polite × Technical Tasks		12.630
		(12.022)
Polite × Editing & Rewriting		19.186
		(13.209)
Polite × Information Seeking		16.760
		(11.003)
Polite × Summarization		2.195
		(12.937)
Polite × Text Generation		-0.492
		(9.789)
Input Tokens	0.920***	0.923***
	(0.060)	(0.060)
Adjusted R-squared	0.0987	0.0988
Number of Observations	31,922	31,922
Note: $^{***}p < 0.01$, $^{**}p < 0.05$, $^{*}p < 0.1$		

Regression results demonstrating that polite prompts reduced output length across all task types, with no significant interaction effects.

To test whether shorter responses from polite prompts reflected reduced quality, the outputs from original and counterfactual prompts were compared for semantic similarity. Using the all-MiniLM-L6-v2 model, each response was embedded in a semantic vector space, and cosine similarity computed between each pair, yielding an average similarity of 0.78, indicating strong alignment in meaning, and suggesting that the content remained consistent even when tone changed.

Stop Words

To understand what kinds of content are reduced in shorter outputs, the most frequently-dropped words were examined. These were found to be common stop-words such as *'have'*, *'more'*, *'where'*, and *'into'*, i.e., terms that serve grammatical rather than semantic roles.

To confirm that token reduction was not driven by loss of meaningful content, stop-words were removed, and phrases of up to four words were analyzed for systematic disappearance; however, no consistent or semantically important patterns were found, suggesting that reductions from polite phrasing were *not* stripping out meaningful or useful content.

Thus, it still seemed that more tokens were being spent on replies to impolite queries than polite ones – like a kind of 'tax' on brusqueness.

Human Study

To test whether output quality was affected by prompt tone, a human evaluation was also conducted, using a random sample of twenty *polite* and twenty *non-polite* prompt pairs.

After excluding prompts on sensitive or technical topics, responses were rated by 401 participants on a seven-point scale. Each participant saw only one response, drawn from one of four conditions: *polite* or *non-polite*, and either *original* or *counterfactual*.

No significant differences were found in perceived quality across any of these conditions. *Polite* and *non-polite* outputs received nearly identical scores, as did *original* and *counterfactual* versions.

The authors assert that these results indicate that the reduction in output tokens was not caused by any loss of quality, but instead by rewording, or through structural shifts that nonetheless preserved meaning.

The cost difference observed in enterprise prompting is therefore unlikely to reflect changes in usefulness or clarity, and the 'tax' is still operative.

Conclusion

Though the new work concentrates on enterprise usage of ChatGPT, lower-tier users are also affected by this syndrome, since even the two entry-level tiers have usage limits; and – presumably – treating ChatGPT gruffly will speed the average user towards the drainage of the day's allocation of tokens.

The new study concentrates on a headline-garnering and much studied open question in human/AI interactions; but the authors emphasize that issues around politeness should be taken as indicators of a possible far deeper well of linguistic eccentricities, as yet undiscovered, that may turn out to be impacting charges for inference.

First published Wednesday, November 19, 2025

Writer on machine learning, domain specialist in human image synthesis. Former head of research content at Metaphysic.ai, until its dissolution into DNEG's Brahma.ai.
Portfolio site: martinanderson.ai
Contact: martin@martinanderson.ai

Discover More