ShareChat: A Dataset of Chatbot Conversations in the Wild

Anonymous ACL submission

Abstract

While academic research typically treats Large Language Models (LLM) as generic text generators, they are distinct commercial products with unique interfaces and capabilities that fundamentally shape user behavior. Current datasets obscure this reality by collecting text-only data through uniform interfaces that fail to capture authentic chatbot usage. To address this limitation, we present **ShareChat**, a large-scale corpus of 142,808 conversations (660,293 turns) sourced directly from publicly shared URLs on ChatGPT, Perplexity, Grok, Gemini, and Claude. ShareChat distinguishes itself by preserving native platform affordances, such as citations and thinking traces, across a diverse collection covering 101 languages and the period from April 2023 to October 2025. Furthermore, ShareChat offers substantially longer context windows and greater interaction depth than prior datasets. To illustrate the dataset's breadth, we present three case studies: a completeness analysis of intent satisfaction, a citation study of model grounding, and a temporal analysis of engagement rhythms. This work provides the community with a vital and timely resource for understanding authentic user-LLM chatbot interactions in the wild. The dataset will be publicly available.

1 Introduction

Conversational Large Language Model (LLM)-based chatbot services have evolved rapidly in the past three years. The first widely adopted general-purpose LLM chatbot, ChatGPT, was launched in November 2022 and has reached more than 700 million weekly active users by mid-2025 (OpenAI Research, 2025). Following this success, many companies quickly released their own chatbot services, which also attracted substantial public attention. For example, Anthropic introduced the Claude families in 2023 (Anthropic, 2023, 2024, 2025), Google deployed the Gemini family of models (Anil et al., 2023), xAI launched Grok as a chatbot integrated with the social media platform X (xAI, 2023), and Perplexity emerged as an answer engine that combines conversational interaction with web search (Perplexity AI, 2024).

Although all of these services are built on text-based LLMs, they differ in interface design, supported features, and safety policies, which in turn shape how users interact with them. For example, Grok can surface live posts from X when providing answers, Claude models are optimized and evaluated for coding, math, and analysis tasks (Anthropic, 2023, 2024), and Perplexity consistently presents responses with explicit source citations (Perplexity AI, 2024). Commercial systems typically undergo continuous reinforcement learning and fine tuning on conversation logs and human feedback (Ouyang et al., 2022; Bai et al., 2022; Chen et al., 2024), which tends to reinforce platform-specific strengths and norms over time.

However, current research often fails to capture the complexity of real-world LLM deployment. While commercial platforms have evolved into complex ecosystems with unique features, the scientific community relies on datasets that homogenize these interactions. As seen in both synthetic (Xu et al., 2024; Li et al., 2025; Ding et al., 2023) and real-world datasets (Zhao et al., 2024; Zheng et al., 2024; Köpf et al., 2023; Zhang et al., 2025; Taori et al., 2023; Conover et al., 2023), current corpora systematically strip away interface context, treating diverse products as identical, generic text boxes. For instance, WildChat (Zhao et al., 2024) compiles about one million conversations between users and ChatGPT collected via an access gateway that offered free usage in exchange for consent to share logs. LMSYS-Chat-1M (Zheng et al., 2024) contains one million conversations with twenty five state of the art models recorded through a single web interface (the Vicuna

Dataset	**#Convs**	**#Users**	**#Turns**	**Avg. Turns**	**#User Tok**	**#Chatbot Tok**	**#Langs**
Existing Public Datasets							
Alpaca	52,002	–	52,002	1.00	$19.67_{\pm 15.19}$	$64.51_{\pm 64.85}$	1
Open Assistant	46,283	13,500	108,341	2.34	$33.41_{\pm 69.89}$	$211.76_{\pm 246.71}$	11
Dolly	15,011	–	15,011	1.00	$110.25_{\pm 261.14}$	$91.14_{\pm 149.15}$	1
ShareGPT	94,145	–	330,239	3.51	$94.46_{\pm 626.39}$	$348.45_{\pm 269.93}$	41
LMSYS-Chat-1M	1,000,000	210,479	2,020,000	2.02	$69.83_{\pm 143.49}$	$215.71_{\pm 1858.09}$	65
Our Multi-Platform Dataset							
Multi-Platform (Total)	**142,808**	–	**660,293**	**4.62**	$135.04_{\pm 1820.88}$	$1,115.30_{\pm 1764.81}$	**101**
Per-Platform Breakdown							
ChatGPT	102,740	–	542,148	5.28	$142.35_{\pm 1191.57}$	$1,230.25_{\pm 2448.38}$	101
Perplexity	17,305	4,763	24,378	1.41	$33.07_{\pm 261.74}$	$573.33_{\pm 932.90}$	45
Grok	14,415	–	53,094	3.69	$179.04_{\pm 6999.90}$	$1,141.74_{\pm 1506.97}$	60
Gemini	7,402	–	36,422	4.92	$184.62_{\pm 1571.62}$	$803.23_{\pm 1609.27}$	47
Claude	946	–	4,251	4.49	$138.67_{\pm 2213.46}$	$576.16_{\pm 1649.61}$	19

Table 1: Comprehensive dataset comparison showing existing corpora, multi-platform aggregate, and per-platform breakdown. Our multi-platform dataset demonstrates superior linguistic diversity (101 languages) and conversation depth. Token statistics computed using Llama-2 tokenizer.

demo and Chatbot Arena). OpenAssistant Conversations (Köpf et al., 2023) provides human generated assistant style dialogues and detailed preference annotations to support alignment research. These datasets are important resources with several limitations:

1. **Artificial Uniformity:** Existing datasets typically collect data via a single, generic interface to standardize the process. This distorts reality. By forcing every model into the same neutral interface, these datasets strip away the unique tools that guide user behavior. Users do not interact with a generic "model" in the real world; they interact with a specific product, and scientific benchmarks should reflect this distinction.
2. **Loss of Non-Textual Context:** In addition, while current datasets are largely limited to plain text, authentic LLM interactions rely on rich structural elements, such as visible thinking traces and embedded source citations, that are often stripped from standard logs. This prevents researchers from studying how critical product features actually influence user behavior and prompting strategies.
3. **Limited Interaction Depth:** Moreover, although some datasets explicitly include multi-turn conversations (Zhao et al., 2024), the average depth of a conversation is still relatively short. In practice, the use of conversational LLMs is much more diverse and complex: users ask follow up questions, change goals, or co-construct content over many turns. Recent work has shown that even state of the art models often get "lost" in multi-turn conversations, with reliability degrading as instructions are refined across turns (Laban et al., 2025a). Given that context length is central to LLM training and performance, longer real user-LLM conversations, extended in both turn count and token length, are valuable both for understanding how user intents and prompts evolve and for investigating when and how models succeed or fail (Laban et al., 2025a).
4. **"Forced" Sharing:** Finally, most datasets rely on participants who know they are being watched, introducing the **Observer Bias** (or Hawthorne Effect) (McCambridge et al., 2014). Users who are aware they are being monitored for research purposes may consciously or unconsciously alter their behavior, which often exhibits higher social desirability compared to authentic, unobserved private usage.

In this study, we address these gaps by constructing a large scale, multi-platform corpus of authentic user-LLM conversations. We compile **142,808 conversations** with more than **660,293 interaction turns** from five widely used platforms: ChatGPT, Perplexity, Grok, Gemini, and Claude, as shown in Tabel 1. All conversations are collected from URLs that users chose to share publicly on the respective platforms, spanning more than thirty months of AI chat usage (April 2023 to October 2025) and covering 101 distinct languages. As SHARECHAT relies on *post-hoc* sharing, it therefore captures authentic usage patterns from a diverse, global user base, rather than simulated prompts or tasks designed by researchers. Each conversation record contains the complete sequence of user and assistant turns,

Feature/Field	ChatGPT	Perplexity	Grok	Gemini	Claude
Textual Content	✓	✓	✓	✓	✓
Source Citations	-	✓	✓	-	-
Thinking Blocks	-	-	✓	-	✓
Code Artifacts	-	-	-	-	✓
Analysis Blocks	-	-	-	-	✓
Turn Timestamps	✓	-	✓	-	-
Model Version	✓	-	✓	✓	-
View/Share Counts	-	✓	-	-	-

Table 2: Feature overview and data extraction capabilities across platforms. Checkmarks indicate feature presence; crosses indicate absence.

along with platform-specific metadata, when available, timestamps, model thinking traces, or source links attached by the interface. Table 2 outlines the distinct data attributes captured across the five platforms, and the details are shown in Appendix A.2. The data collection is done under IRB approval (#28569).

To demonstrate the specific utility of SHARECHAT, besides describing the basic statistics of the corpus, we present three illustrative case studies. First, we perform conversation completeness analysis to assess if user intentions are fully addressed, leveraging SHARECHAT's high interaction depth to capture complex goal resolutions. Second, we conduct source domain analysis to evaluate how models ground claims, utilizing preserved non-textual metadata to study citation behaviors. Third, we execute temporal analysis using turn-level timestamps, to reveal authentic interaction rhythms and engagement patterns free from observer bias.

Our contributions are twofold. First, we construct the first large-scale corpus of user-LLM interactions that preserves native affordances of multiple platforms, with substantially longer conversations on average and lower levels of toxic content than prior corpora. Second, we provide three concrete case studies, which together demonstrate how the dataset can be used to study user interaction behavior in depth.

2 Data Collection

For the five different platforms, we develop systematic, platform-specific data extraction pipelines to collect conversation content, metadata, and platform-specific features from publicly accessible shared conversation URLs found on the Internet. Although we find that such URLs are often shared on social media platforms such as X (formerly Twitter), Reddit, and Discord, exhaustively monitoring these individual ecosystems presents significant scalability challenges. To tackle this, we utilize Internet archival services, specifically the Internet Archive (Wayback Machine), to search for and extract URLs matching specific patterns (e.g., *"chatgpt.com/share/*"* for ChatGPT). This archival-based discovery method is a well-established strategy for aggregating public digital resources, previously employed to compile datasets for web-scale text (Raffel et al., 2020), historical news archives (Hamborg et al., 2017), and corporate disclosure analysis (Boulland et al., 2024). For each platform, data collection covers a distinct time window, as summarized in Appendix A.1.

For every shared conversation page, we use automated browser control with Selenium to interact with the page, retrieve the rendered HTML, and parse it into a structured JSON representation. The resulting records include the ordered sequence of conversation turns, user prompts, model responses, and platform-specific metadata. When certain elements were only visible after user interaction, we script the corresponding actions. For example, to collect the thinking content of Claude, we first locate the thinking bar element and then trigger a click on it when the content is only accessible through interaction.

Filtering of Identifiable Information. We prioritize user privacy by anonymizing personally identifiable information (PII). We use Microsoft's Presidio[1] as the framework, Spacy (Honnibal et al., 2020) for Named Entity Recognition (NER), and custom rules to identify and remove PII across various data types, such as names, phone numbers, emails, credit cards, and URLs, in multiple languages including English, Chinese, Russian, French, Spanish, German, Portuguese, Italian, Japanese, and Korean.

3 Preliminary Analysis

The final dataset contains conversations from five platforms, featuring a wide range of languages, diverse user prompts and longer conversations with less toxic content compared to other datasets. In this section, we describe the basic statistics of the

[1] https://microsoft.github.io/presidio/

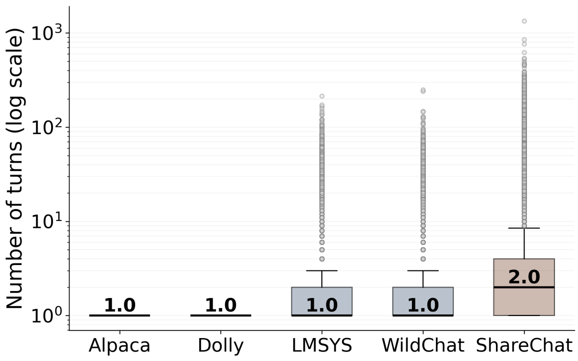


Figure 1: Turn length distribution across datasets

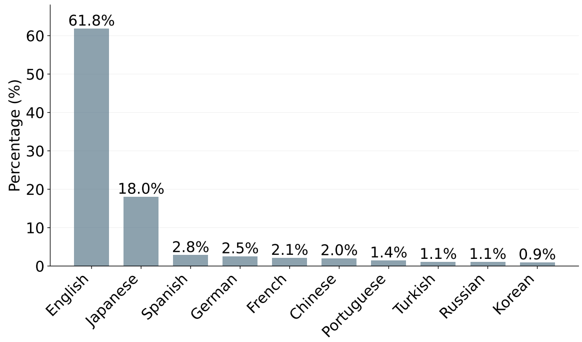


Figure 2: Distribution of the top 10 languages in SHARECHAT.

dataset, and the results of our toxicity analysis and topic distribution.

3.1 Basic Statistics

In total, SHARECHAT has **142,808 conversations** with more than **660,293 interaction turns**. Table 1 compares ShareChat with existing corpora across key metrics: turn count, token length, and language diversity. The language detection was performed at message-level using lingua-py and then combined into conversation-level statistics. Token counts were computed using the Llama-2 tokenizer for consistent cross-platform comparison (Zhao et al., 2024).

In summary, **SHARECHAT distinguishes itself by extending the scope of analysis along two critical dimensions: interaction depth and content volume.** First, conversations are substantially longer, averaging 4.62 turns with a heterogeneous distribution that consistently exceeds existing public corpora (Figure 1 and Appendix A.3). Second, model responses are significantly denser, with a mean of 1,115.30 tokens compared to just 215.71 for LMSYS-Chat-1M, suggesting the presence of richer and more substantive problem-solving.

SHARECHAT also demonstrates greater linguistic diversity by spanning 101 distinct languages, whereas the most diverse competitor covers only 65. For languages covered, as shown in Figure 2, English accounts for the clear majority of conversations across all platforms, with Japanese as the second most frequent language and all other languages each contributing a much smaller share. Compared to previous datasets, SHARECHAT has a more balanced language distribution.

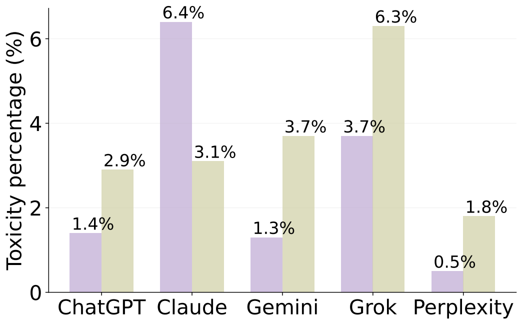
3.2 Toxicity Analysis

To assess toxicity in SHARECHAT, we employ both Detoxify (Hanu and Unitary team, 2020) and OpenAI Moderation API[2] to detect toxic content in the conversations, following the approach used in WildChat (Zhao et al., 2024). Detoxify is a pre-trained multilingual toxicity classification model that computes toxicity scores across seven dimensions, while OpenAI Moderation provides commercial-grade content filtering. Given the language coverage limitations of Detoxify, we retain only conversational turns in languages explicitly supported by the model, with more details shown in Appendix A.4. For Detoxify, we apply the same classification threshold as WildChat, marking messages as toxic when the toxicity score exceeds 0.1, while OpenAI Moderation uses its internal flagging mechanism to identify toxic content.

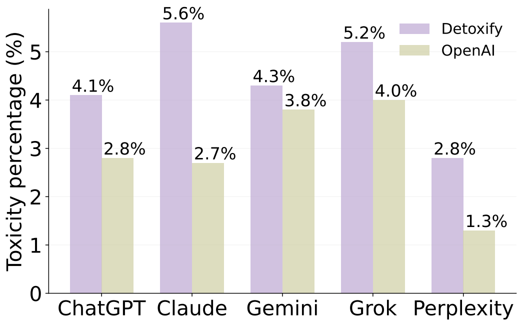
The results for turn-level toxicity are shown in Figure 3, which reveal substantial differences between the two detection methods, consistent with limited agreement observed in prior work (Zhao et al., 2024). Furthermore, we find a statistically significant positive rank-order correlation between user and LLM toxicity rates within platforms (Spearman's $\rho = 0.90, p = 0.037$ for Detoxify; $\rho = 0.88, p = 0.049$ for OpenAI). This suggests a "mirroring" effect where platforms receiving more toxic user inputs tend to generate more toxic outputs. For instance, Claude ranks highest in both user (5.6%) and model toxicity (6.4%) under Detoxify.

Most importantly, **SHARECHAT contains substantially less toxic content compared to existing benchmarks**. Our overall user toxicity rates of 4.1% via Detoxify and 2.9% via OpenAI are considerably lower than both WildChat at 6.05% and LMSYS-Chat-1M at 3.08% using the OpenAI method (Zhao et al., 2024). Similarly, our LLM tox-

[2]https://platform.openai.com/docs/guides/moderation



(a) LLM turn-level toxicity



(b) User turn-level toxicity

Figure 3: Turn-level toxicity comparison by platform. These bar charts compare toxicity detection rates at the individual message level across five AI platforms using two different detection methods: Detoxify and OpenAI Moderation API. The panel (a) shows LLM message toxicity rates, while panel (b) shows user response toxicity rates.

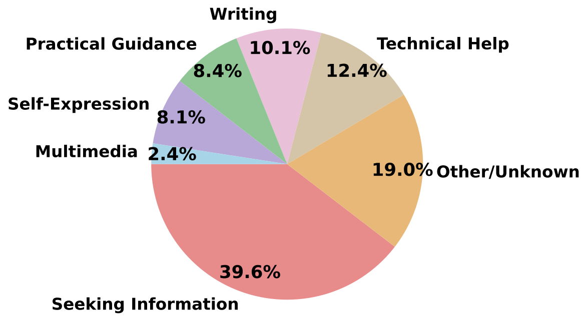


Figure 4: Average topic distribution of user requests across five platforms.

icity rates of 1.6% via Detoxify and 3.2% via OpenAI are lower than the 5.18% reported by WildChat and comparable to the 4.12% in LMSYS-Chat-1M.

3.3 Topic Analysis

We classify each user message into one of 24 predefined topic categories using multilingual LLMs Llama-3.1-8B-Instruct [3] under a few-shot prompting setup. To enable large-scale processing, we run inference with 4-bit quantization, batch size of 32. Model outputs are normalized through a post-processing pipeline to handle formatting variants and stray meta-responses, ensuring reliable mapping to our fixed taxonomy. For downstream analyses, the 24 fine-grained categories are further consolidated into seven higher-level groups to support both granular and coarse-grained comparisons across platforms. The 7 high-level categories of user requests are plotted in Figure 4.

Overall, the averaged distribution reveals that **information seeking remains the core driver of LLM interactions, with creative, technical, and advisory uses forming important but secondary dimensions of user behavior**. Specifically, we observe a strong central tendency in how users engage with LLMs. *Seeking Information* overwhelmingly dominates usage, comprising 39.6% of all requests, underscoring that fact-oriented question-answering remains the primary interaction paradigm. The second-largest category, *Other/Unknown* (19.0%), reflects a substantial proportion of prompts that are either ambiguous, exploratory, or insufficiently specified for reliable categorization. *Technical Help* (12.4%) and *Writing* (10.1%) constitute the next major cluster of user intent, highlighting the continued reliance on LLMs for procedural support and text generation tasks. *Practical Guidance* (8.4%) and *Self-Expression* (8.1%) appear at comparable frequencies, indicating moderate but consistent engagement with step-by-step advice and personal or expressive communication. Finally, *Multimedia* accounts for only 2.4% of requests, likely reflecting limited multimodal capabilities at the time of data collection.

Also, we find that users employ platforms for distinct roles, such as using Perplexity for "seeking information" versus Claude for "technical thinking". Detailed prompts, category definitions, classification statistics and platform-specific results are provided in the Appendix A.5.

4 Representative Analyses

4.1 Conversation Completeness Analysis

To assess how comprehensively LLMs fulfill user needs in multi-turn dialogues, we develop an automated completeness analysis pipeline that evaluates whether each user intention across a conversation is adequately addressed by the assistant. Our anal-

[3] https://huggingface.co/meta-llama/Llama-3.1-8B

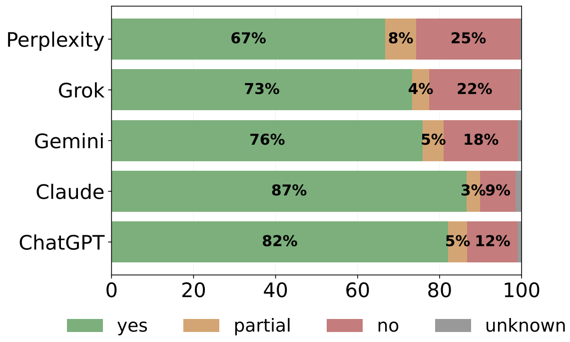


Figure 5: Verdict distribution by platform. It shows the proportion of user intentions receiving complete, partial, incomplete, or unknown verdicts at the intention level.

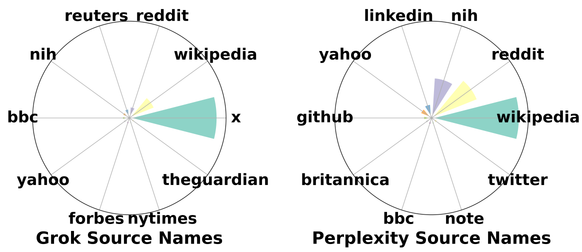


Figure 6: Response source top frequency rose graph presents the rose plots of the top 10 most frequent source domains, highlighting X as Grok's dominant source and Wikipedia as Perplexity's leading source.

ysis reveals distinct performance across platforms, with ChatGPT and Claude demonstrating the highest consistency in fully satisfying user intentions, while Gemini, Grok and Perplexity exhibit significantly wider variance in conversational quality across multi-turn interactions.

To assess the conversation completeness, we implement a three-stage pipeline using Qwen3-8B (Yang et al., 2025). First, we extract distinct user intentions from the conversation history. Second, we classify the resolution of each intention as *complete*, *partial*, or *incomplete*. Finally, we compute a conversation-level completeness score by weighting these verdicts. We sample 100 conversations (20 per platform) for human evaluation. Our coding revealed that 85.1% of user intentions are correctly identified, and among these, 92.5% are correctly labeled. Through this annotation scheme, the number of extracted intentions per conversation varies systematically across platforms, with ChatGPT and Claude conversations containing a median of 2 intentions, while Gemini, Grok, and Perplexity show a median of 1 intention per conversation. Full prompts and implementation details are provided in Appendix A.6.

Different platforms reveal substantial variation in how effectively they address user intentions in multi-turn conversations. ChatGPT and Claude exhibit the highest rates of complete verdicts in Figure 5, with both platforms fully resolving the majority of user intentions. Gemini and Grok show more balanced distributions between complete and partial verdicts, with a substantial proportion of intentions receiving incomplete verdicts. Perplexity demonstrates an intermediate pattern with the highest rate of partial verdicts across all platforms.

These findings highlight the **dataset's structural diversity, spanning single-turn queries to complex, multi-intention sessions.** The prevalence of partial verdicts reveals that real-world goals often remain incompletely satisfied, which is a dynamic obscured by benchmarks prioritizing single-turn correctness. By providing these completeness labels, SHARECHAT enables the development of metrics aligned with user satisfaction and offers rich signals for reinforcement learning.

4.2 Source Grounding Analysis

We observe distinct information retrieval paradigms between the two search-enabled platforms. As detailed in Table 2, external grounding is a core feature for both systems: the Grok dataset contains 14,415 conversations with 8,242 (57.18%) including sources, while the Perplexity dataset includes 17,305 conversations with 8,545 (49.38%) containing sources.

As illustrated in Figure 6, Grok exhibits a hyper-concentrated reliance on social data. Its source distribution is heavily skewed toward the X platform, which accounts for 40,624 references, nearly triple the citations for Wikipedia (12,507). This suggests a design optimized for real-time, social-media-driven context, potentially offering higher timeliness but introducing risks related to information stability and bias.

In contrast, Perplexity demonstrates a diversified, encyclopedic retrieval strategy. Its distribution is significantly less concentrated, with English Wikipedia as the leading source (2,919 citations), followed by community forums like Reddit (1,642) and authoritative scientific repositories like NIH (1,339). Furthermore, as detailed in Appendix A.7, Perplexity supports much deeper research sessions, with some conversations integrating over 1,000 distinct sources, whereas Grok typically cites fewer than 20 sources per interaction.

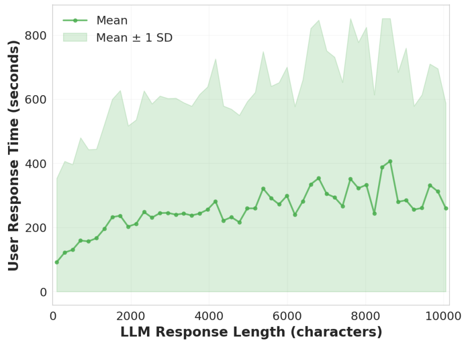
Taken together, these plots demonstrate that the **two platforms employ fundamentally distinct information retrieval strategies, prioritizing different types of external evidence to ground their generated responses**.

4.3 Timestamp Analysis

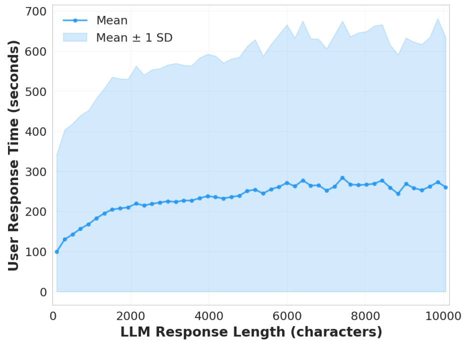
Timestamp metadata is available for ChatGPT and Grok, covering nearly all turns (99.97% for ChatGPT and 100% for Grok). For the small fraction of missing timestamps in ChatGPT data, we employ linear interpolation between known points where feasible, and dropped affected turns when interpolation was not possible. We verify that results remained stable across different handling strategies, confirming that the missing data does not bias our findings. After removing extreme outliers above the 99th percentile, we observe systematic differences between platforms. ChatGPT exhibits longer mean user response times than Grok (1,580 vs. 931 seconds), while producing shorter mean LLM response times (18.4 vs. 24.6 seconds).

Then, we examine how response times evolve over the course of a conversation by correlating turn position with response latency. ChatGPT exhibits a significant negative correlation between turn index and LLM response time (Pearson $r = -0.238$; Spearman $\rho = -0.427$), indicating progressively faster responses as conversations lengthen, consistent with caching, adaptation, or context optimization effects. In contrast, Grok shows a significant positive correlation (Pearson $r = 0.315$; Spearman $\rho = 0.254$), suggesting increasing computational overhead as context accumulates. User response times show minimal linear association with turn position on both platforms. These divergent temporal dynamics highlight fundamental architectural differences in how the two systems handle extended interactions.

To evaluate whether verbosity alone provides a reliable predictor of user processing time, we additionally examine correlations between LLM message length and subsequent user response latency. Figure 7 shows that for both Grok and ChatGPT, mean user response time increases with LLM response length, with the steepest growth at shorter outputs and a plateau emerging beyond approximately 4–6k characters. The shaded regions (± 1 SD) reveal substantial variability across users. Amongst 6k to 10k the variance further widens and this suggests increased heterogeneity in how users engage with lengthy outputs. This trend suggests that while longer responses are associated with higher average response times in aggregate, length alone is insufficient to predict individual-level latency, with more details in Appendix A.8.



(a) Grok



(b) ChatGPT

Figure 7: Binned mean user response time as a function of LLM response length. Solid lines denote mean response time across users, while shaded regions indicate ± 1 standard deviation.

5 Discussion

The findings presented in this study offer several important implications for understanding human-AI interaction and guiding future research on conversational LLMs. While we only have demonstrated three concrete representative analyses, the dataset's utility extends well beyond these applications.

Complex Interaction Dynamics in Extended Contexts. SHARECHAT's longer conversation contexts, averaging 4.62 turns compared to 2.02 in LMSYS-Chat-1M (Zheng et al., 2024), provide a critical resource for studying phenomena that emerge only over extended interactions. Prior work has shown that model reliability can degrade as instructions are refined across turns (Laban et al., 2025b), yet most existing corpora offer limited opportunities to investigate such dynamics due to their predominantly single-turn structure. The presence of conversations extending into dozens of turns, combined with the diversity of intention counts per conversation, enables researchers to examine how user needs evolve, how models maintain coherence over long contexts, and when breakdowns occur. SHARECHAT makes it possible to study these partial successes systematically, revealing where assistants provide useful but insufficient information and how users adapt their follow-up requests in response. This diversity provides the training signals necessary for models to learn adaptive strategies, specifically, when to shift focus, how to revisit unresolved intentions, and how to maintain coherence as user goals evolve.

Platform-Dependent Usage and Performance Variability. Unlike the "universal assistant" assumption often found in general benchmarks, our cross-platform analysis reveals that user behavior is highly sensitive to the specific design and functional positioning of each platform. Our topic distribution analysis demonstrates that users align their requests with perceived platform strengths, such as using Perplexity for information retrieval versus Claude for technical help. This functional specialization is mirrored in conversation completeness that while ChatGPT and Claude exhibit narrow, concentrated completeness scores, Gemini and Grok show substantial variability. Notably, the high frequency of "partial" verdicts in Perplexity interactions reflects its distinct role as a search engine, where outputs often serve as intermediate steps rather than final answers. Researchers can use SHARECHAT to investigate these specialized performance profiles, moving beyond generic comparisons to determine which features drive reliability for specific user intents.

Evidence-Based Generation and Temporal Metadata. The preservation of rich metadata enables future research on retrieval-augmented generation (RAG) and interaction pacing in naturalistic settings. Our source content analysis reveals pronounced differences in how platforms integrate external evidence. Grok relies overwhelmingly on concentrated source types, while Perplexity draws from a diversified set of domains. These patterns have implications for understanding how users interpret and rely on cited evidence, and how platform affordances shape trust and verification behaviors. Additionally, the availability of turn-level timestamps enables the investigation of conversational rhythm and interaction pacing, offering a temporal dimension of user experience. Our analysis exposes contrasting architectural dynamics: while ChatGPT demonstrates progressively lower latency as conversations extend over multiple turns, Grok exhibits an inverse trend, with users experiencing slower generation speeds as context length grows.

Self-Selection and Safety in Public Sharing. Finally, the substantially lower toxicity rates in our corpus compared to existing benchmarks, for both user messages and LLM responses, likely reflect the self-selection inherent in public sharing. Users who choose to share conversations may filter for interactions they consider appropriate or valuable, producing a corpus that captures more constructive usage patterns. This suggests that SHARECHAT offers a cleaner, high-quality resource for studying successful task completion and constructive human-AI collaboration.

6 Conclusion

In this study, we address the limitations of existing datasets by constructing SHARECHAT, a large-scale, multi-platform corpus of user-LLM conversations. By compiling more than 140,000 publicly shared interactions from five distinct platforms, we provide a resource that preserves the native metadata and diverse interaction designs of the current AI ecosystem. The dataset's basic statistics show that the self-selection process inherent in public sharing leads to a corpus with significantly longer conversation contexts and lower toxicity.

We further illustrate the value of this data through three specific analyses. First, our conversation completeness analysis indicate that user intentions are more frequently satisfied in general purpose models like ChatGPT and Claude, reflecting their emphasis on broad capabilities. Second, our source content analysis uncovers the distinct information retrieval strategies employed by Grok and Perplexity. Third, our temporal analysis suggests that ChatGPT and Grok exhibit contrasting temporal dynamics.

Collectively, these findings highlight pronounced differences in user behavior that align with the functional emphasis of each platform, underscoring the importance of studying LLM capabilities within their specific contexts. SHARECHAT not only provides a testbed for evaluating model reliability across turns, but also offers rich supervision for training models to handle complex, multi-turn user needs in different settings.

Limitations

Our study is subject to several limitations that should be considered when interpreting the results.

Selection Bias. First, the reliance on publicly shared URLs introduces a trade-off between two forms of bias. On one hand, this approach mitigates the **Observer Bias** which we talked about in Section 1. On the other hand, it introduces **Self-Selection Bias** (Heckman, 1979), because users voluntarily choose which conversations to generate links for, the corpus is likely skewed towards interactions they deem successful, interesting, or novel. Therefore, SHARECHAT may underrepresent mundane utility tasks or failed interactions where the model performed poorly, as users are less motivated to share such content publicly (Hu et al., 2017).

Platform Imbalance. Second, the dataset exhibits significant imbalance across platforms, reflecting current disparities in market share. ChatGPT accounts for over 70% of the collected conversations, whereas Claude represents less than 1%. Although we continue to expand the collection pipeline, researchers should exercise caution when performing direct cross-platform comparisons and may need to employ stratified sampling to account for volume differences.

Proxy Nature of Automated Metrics. Third, while our conversation completeness analysis and topic analysis utilize robust LLM-based evaluation, it remains a proxy metric. Automated judges provide a scalable approximation of user satisfaction but cannot perfectly capture the nuanced, subjective quality of human preference. These results should be interpreted as strong signals of structural completeness rather than absolute ground-truth measurements of user happiness. Future research is needed to validate these automated metrics against human judgment and to fully explore the evolving longitudinal dynamics of user behavior.

Preliminary Scope of Analysis. Finally, the specific analyses presented in this work, such as the source domain distribution and temporal engagement patterns, are intended as illustrative representative analyses to demonstrate the dataset's utility. They are not exhaustive sociological studies. Future research is needed to fully explore these dimensions, particularly how these patterns evolve longitudinally as platform features change.

Ethical Considerations

Data Provenance and Consent. The data collection methodology employed in this study strictly adheres to ethical standards regarding public data usage. The corpus consists exclusively of conversations sourced from URLs that users voluntarily generated and shared via the platforms' native sharing features. By actively generating a shareable link, users establish a clear intent for public visibility, distinguishing these artifacts from private chat logs or leaked data. We did not attempt to bypass authentication walls, scrape private user accounts, or access data that was not publicly indexed.

Privacy Preservation and De-Identification. Despite the public provenance of the data, we prioritized the protection of user privacy through a rigorous de-identification pipeline designed to mitigate risks associated with inadvertently exposed sensitive information. First, we utilized automated NER and regular expression matching to detect and redact PII, including real names, email addresses, phone numbers, and IP addresses, for different languages. In addition, we do not store or release original platform-specific user IDs or usernames. All user identifiers were cryptographically hashed and converted into internal project identifiers during the initial processing stage. Finally, the analyses presented in this paper, like the toxicity assessments and completeness scoring, are conducted solely on aggregated statistics. No individual user's conversation history is singled out for qualitative scrutiny in a manner that could lead to re-identification or personal embarrassment.

Content Safety and Intended Use. Researchers must be aware that SHARECHAT reflects real-world interactions and, consequently, contains content that may be offensive, biased, or toxic, as quantified in our toxicity analysis. We release this data specifically for research purposes, such as understanding human-AI interaction dynamics, studying safety alignment, and evaluating model performance, and not for training user-facing production models without further filtering. Users of SHARECHAT are required to adhere to the terms of use, which strictly prohibit attempts to de-anonymize users or leverage the data for malicious surveillance. By limiting collection to publicly accessible links and proactively scrubbing personal data, this work aims to balance the scientific value of ecological data with harm minimization.

References

Rohan Anil and 1 others. 2023. Gemini: A family of highly capable multimodal models. *Preprint*, arXiv:2312.11805.

Anthropic. 2023. Model card and evaluations for claude 2. Technical report, Anthropic. Model card.

Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku. `https://www.anthropic.com/claude-3-model-card`. Model card.

Anthropic. 2025. System card: Claude opus 4 & claude sonnet 4.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jacob Sanders, and 1 others. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *Preprint*, arXiv:2204.05862.

Romain Boulland, Thomas Bourveau, and Matthias Breuer. 2024. Corporate websites: A new measure of voluntary disclosure. *Journal of Accounting Research*, 62(4):1297–1353.

Lingjiao Chen, Matei Zaharia, and James Zou. 2024. How is chatgpt's behavior changing over time? *Harvard Data Science Review*, 6(2).

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. Free dolly: Introducing the world's first truly open instruction-tuned llm.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3029–3051.

Felix Hamborg, Norman Meuschke, Corinna Breitinger, and Bela Gipp. 2017. Automated identification of media bias in news articles: an interdisciplinary approach. In *Proceedings of the 17th ACM/IEEE Joint Conference on Digital Libraries*, pages 396–397.

Laura Hanu and Unitary team. 2020. Detoxify. `https://github.com/unitaryai/detoxify`.

James J. Heckman. 1979. Sample selection bias as a specification error. *Econometrica*, 47(1):153–161.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength natural language processing in python.

Nan Hu, Noi Sian Koh, and Srinivas K Reddy. 2017. On the readabilty of sentiment analysis: The impact of selection bias in online reviews. In *Proceedings of the 50th Hawaii International Conference on System Sciences*.

Andreas Köpf, Yannic Kilcher, Dimitri Von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, and 1 others. 2023. Openassistant conversations-democratizing large language model alignment. *Advances in neural information processing systems*, 36:47669–47681.

Philippe Laban, Hiroaki Hayashi, Yingbo Zhou, and Jennifer Neville. 2025a. Llms get lost in multi turn conversation.

Philippe Laban, Hiroaki Hayashi, Yingbo Zhou, and Jennifer Neville. 2025b. Llms get lost in multi-turn conversation. *Preprint*, arXiv:2505.06120.

Jijie Li, Li Du, Hanyu Zhao, Bo-wen Zhang, Liangdong Wang, Boyan Gao, Guang Liu, and Yonghua Lin. 2025. Infinity instruct: Scaling instruction selection and synthesis to enhance language models. *arXiv preprint arXiv:2506.11116*.

Jim McCambridge, John Witton, and Diana R Elbourne. 2014. Systematic review of the hawthorne effect. *Journal of Clinical Epidemiology*, 67(3):267–277.

OpenAI Research. 2025. Chatgpt usage and adoption patterns at work. Technical report, OpenAI. Technical report.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35.

Perplexity AI. 2024. What is perplexity? `https://www.perplexity.ai/help-center/en/articles/10352155-what-is-perplexity`. Help center article.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Stanford alpaca: An instruction-following llama model.

xAI. 2023. Announcing grok. Accessed: 2025-10-15.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Yuntian Deng, Radha Poovendran, Yejin Choi, and Bill Yuchen Lin. 2024. Magpie: Alignment data synthesis from scratch by prompting aligned llms with nothing. *arXiv preprint arXiv:2406.08464*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Yuanming Zhang, Yan Lin, Arijit Khan, and Huaiyu Wan. 2025. Large language model prompt datasets: An in-depth analysis and insights. *arXiv preprint arXiv:2510.09316*.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. 2024. Wildchat: 1m chatgpt interaction logs in the wild. *arXiv preprint arXiv:2405.01470*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Tianle Li, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zhuohan Li, Zi Lin, Eric P Xing, and 1 others. 2024. Lmsys-chat-1m: A large-scale real-world llm conversation dataset. *Proceedings of the International Conference on Learning Representations*.

A Appendix

A.1 Data collection

Table 3 details the target URLs used for extraction and the data collection timeframe for each platform.

A.2 Platform-Specific Interaction and Metadata Attributes

Different platforms also reflect distinct design philosophies, ChatGPT and Gemini offer general-purpose conversational assistance. Perplexity focuses on search-oriented information retrieval. Grok targets social media-integrated question answering. Claude specializes in technical and reasoning support. These structural differences align with intended representative analyses and shape the types of user interactions preserved in SHARECHAT.

As shown in Table 2, while ChatGPT and Gemini follow standard conversational formats, other platforms expose specialized content structures. Claude is unique in preserving "thinking blocks" and versioned code artifacts, enabling the analysis of intermediate reasoning and iterative coding processes. In contrast, Perplexity and Grok structure responses around retrieval. Perplexity organizes content into answers, sources, and images with inline citations, while Grok features a dedicated pane for web and social media sources (X posts). These structural differences allow for distinct lines of inquiry: technical workflow analysis for Claude versus citation accuracy and source integration studies for Perplexity and Grok.

The granularity of metadata also dictates the scope of possible temporal analysis. ChatGPT and Grok are the only platforms providing turn-level timestamps, which are essential for modeling conversation rhythm and response latency. ChatGPT further enriches this with model version identifiers (e.g., `gpt-4`). Conversely, Perplexity lacks turn-level timing but offers unique engagement metrics, including view counts, share counts, and last-updated timestamps, effectively treating conversations as evolving public artifacts rather than static logs.

A.3 Turn length Distribution

As shown in Figure 8, both ChatGPT and Claude showing median turn lengths of 2.0, while Gemini, Grok, and Perplexity each maintain a median of 1.0 but with substantially higher variability. The presence of pronounced outliers extending into the hundreds of turns indicates substantial diversity in conversational depth, capturing both brief exchanges and extended dialogues that reflect naturalistic user interactions across platforms.

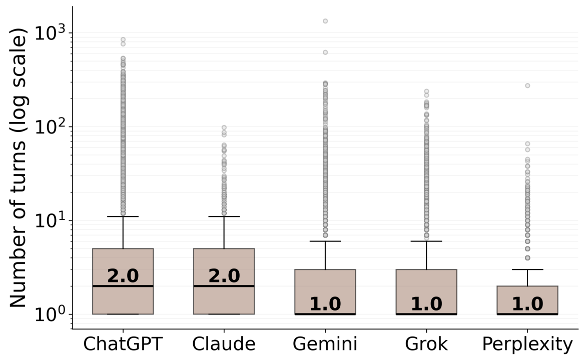


Figure 8: Turn Length Distribution Across collected platforms

A.4 Toxicity Analysis Results

Leveraging the multi-label capabilities of the Detoxify model trained on the Jigsaw Toxic Comment Classification dataset, we analyzed six distinct dimensions of harmful content: **toxicity**, which is a broad category for rude or disrespectful comments, **severe toxicity**, **obscenity**, **threat**, **insult**, and **identity attack**. After filtering conversations supported by the model, resulting in a filtered dataset of 104,107 (72.9% of the original corpus) across all five platforms. Table 4 presents the mean probability scores for each of these six dimensions by platform. The final column, "Ratio Toxic," reports the percentage of turns where the general toxicity score exceeded the standard threshold of 0.1.

The results reveal substantial variation across platforms, with Claude exhibiting the highest prevalence of general toxicity with a mean score of 0.0407, followed by Grok at 0.0258, Gemini at 0.0154, ChatGPT at 0.0146, and Perplexity at 0.0088. Table 5 further summarizes these patterns by role, demonstrating that user contributions consistently scored higher than LLM-generated content across the toxicity dimension with 0.0252 versus 0.0168, as well as in specific categories like threat and identity attack. Conversely, LLM responses showed slightly higher mean scores in the obscene dimension, measuring 0.0136 versus 0.0113.

The conversation-level results are shown in Table 6. It was found that toxicity rates vary substan-

Platform	Share URL Format	Collection Period
ChatGPT	chatgpt.com/share/*	May 2023 to Aug 2025
Perplexity	perplexity.ai/search/*	Apr 2023 to Oct 2025
Grok	grok.com/share/*	Dec 2024 to Oct 2025
Gemini	gemini.google.com/share/*	Apr 2024 to Sep 2025
Claude	claude.ai/share/*	Not available

Table 3: Supported AI chat platforms with share URL formats, primary design focus, and collection timeframes

tially by detection method, with 6.8% to 9.7% of conversations containing toxic user content and 3.0% to 5.8% containing toxic LLM responses when aggregated across platforms. This higher sensitivity of Detoxify compared to OpenAI is consistent with the patterns observed at the turn level. Furthermore, consistent with the turn-level analysis of platform trends, Perplexity exhibited the lowest prevalence of toxic conversations at roughly 3% to 4%, while Grok generally displayed the highest rates ranging from 11% to 13%. This consistency across both levels of analysis confirms that the observed differences are stable and likely stem from the distinct ways each platform is designed and used.

A.5 Detailed Analysis of User Request Topic Distribution

Number of Topics. Our taxonomy comprises seven high-level topic categories: *Multimedia*, *Other/Unknown*, *Practical Guidance*, *Seeking Information*, *Self-Expression*, *Technical Help*, and *Writing*. These categories provide a compact yet expressive organization of user intents, enabling consistent comparative analysis across datasets while preserving sufficient semantic granularity for downstream interpretation.

Discovery Prompt. We provide below the prompt used to discover user-request topics.

```
### SYSTEM ROLE
You are a conversation topic classifier. Output only the
      category name, nothing else.

### INSTRUCTIONS
1. Identify Distinct Goals: Focus on information seeking,
      task requests, or problem-solving goals.
2. Maintain Order: The first item in your list must
      correspond to the user's first real request, and so
      on.
3. Ignore Noise: Skip purely social turns (e.g., "Hello",
      "Thank you", "Okay") unless they are the only
      message.

You are an internal tool that classifies a message from a
      user to an AI chatbot, based on the context of the
      previous messages before it. Based on the last user
      message of this conversation transcript and taking
      into account the examples further below as guidance,
      please select the capability the user is clearly
```

```
      interested in, or `other` if it is clear but not in
      the list below, or `unclear` if it is hard to tell
      what the user even wants:
- edit_or_critique_provided_text: Improving or modifying
      text provided by the user.
- argument_or_summary_generation: Creating arguments or
      summaries on topics not provided in detail by the
      user.
- personal_writing_or_communication: Assisting with
      personal messages, emails, or social media posts.
- write_fiction: Crafting poems, stories, or fictional
      content.
- how_to_advice: Providing step-by-step instructions or
      guidance on how to perform tasks or learn new skills.
- creative_ideation: Generating ideas or suggestions for
      creative projects or activities.
- tutoring_or_teaching: Explaining concepts, teaching
      subjects, or helping the user understand educational
      material.
- translation: Translating text from one language to
      another.
- mathematical_calculation: Solving math problems,
      performing calculations, or working with numerical
      data.
- computer_programming: Writing code, debugging,
      explaining programming concepts, or discussing
      programming languages and tools.
- purchasable_products: Inquiries about products or
      services available for purchase.
- cooking_and_recipes: Seeking recipes, cooking
      instructions, or culinary advice.
- health_fitness_beauty_or_self_care: Seeking advice or
      information on physical health, fitness routines,
      beauty tips, or self-care practices.
- specific_info: Providing specific information typically
      found on websites, including information about
      well-known individuals, current events, historical
      events, and other facts and knowledge.
- greetings_and_chitchat: Casual conversation, small talk,
      or friendly interactions without a specific
      informational goal.
- relationships_and_personal_reflection: Discussing
      personal reflections or seeking advice on
      relationships and feelings.
- games_and_role_play: Engaging in interactive games,
      simulations, or imaginative role-playing scenarios.
- asking_about_the_model: Questions about the AI model's
      capabilities or characteristics.
- create_an_image: Requests to generate or draw new visual
      content based on the user's description.
- analyze_an_image: Interpreting or describing visual
      content provided by the user, such as photos,
      charts, graphs, or illustrations.
- generate_or_retrieve_other_media: Creating or finding
      media other than text or images, such as audio,
      video, or multimedia files.
- data_analysis: Performing statistical analysis,
      interpreting datasets, or extracting insights from
      data.
- unclear: If the user's intent is not clear from the
      conversation.
- other: If the capability requested doesn't fit any of
      the above categories.

Only reply with one of the capabilities above, without
      quotes and as presented (all lower case with
      underscores and spaces as shown). If the
      conversation has multiple distinct capabilities,
      choose the one that is the most relevant to the LAST
      message in the conversation.

Examples:
```

Platform	n	Retention %	Toxicity	Severe Tox.	Obscene	Threat	Insult	Identity Attack
Claude	7,455	87.7	0.0407	0.0017	0.0297	0.0014	0.0127	0.0009
Grok	82,689	77.9	0.0258	0.0011	0.0115	0.0015	0.0118	0.0022
Gemini	52,964	72.8	0.0154	0.0010	0.0099	0.0017	0.0098	0.0019
ChatGPT	761,905	71.7	0.0146	0.0006	0.0070	0.0010	0.0075	0.0015
Perplexity	44,474	81.8	0.0088	0.0005	0.0040	0.0007	0.0051	0.0014

Table 4: Toxicity dimensions by platform (Mean Scores - Detoxify-Supported Languages Only)

Metric	N	Retention %	Toxicity	Severe Tox.	Obscene	Threat	Insult	Identity Attack
All	949,487	72.9	0.0211	0.0010	0.0124	0.0013	0.0094	0.0016
llm	471,020	72.9	0.0168	0.0011	0.0136	0.0006	0.0079	0.0010
user	478,467	72.9	0.0252	0.0009	0.0113	0.0019	0.0108	0.0021

Table 5: Combined toxicity summary across all platforms

Role	ChatGPT		Claude		Gemini		Grok		Perplexity		All Platforms	
(%)	Detox	Open	Detox	Open	Detox	Open	Detox	Open	Detox	Open	Detox	Open
user	10.7	7.5	10.2	6.1	10.1	7.9	11.0	7.1	3.7	1.7	9.7	6.8
llm	3.1	6.0	4.8	5.2	2.9	6.3	5.2	8.1	0.7	2.2	3.0	5.8
All	11.6	10.2	11.1	7.7	10.6	10.3	12.9	11.0	3.9	3.0	10.6	9.4

Table 6: Conversation-level toxicity percentage comparison: Detoxify vs OpenAI moderation

```
- edit_or_critique_provided_text:
"Help me improve my essay, including improving flow and
      correcting grammar errors."
"Please shorten this paragraph."
"Can you proofread my article for grammatical mistakes?
"Here\'s my draft speech; can you suggest enhancements?
"Stp aide moi àcorriger ma dissertation."

- argument_or_summary_generation:
"Make an argument for why the national debt is important.
"Write a three-paragraph essay about Abraham Lincoln."
"Summarize the Book of Matthew."
"Provide a summary of the theory of relativity."
"R'ediger un essai sur la politique au Moyen-Orient."

personal_writing_or_communication:
"Write a nice birthday card note for my girlfriend."
"What should my speech say to Karl at his retirement
      party?"
"Help me write a cover letter for a job application."
"Compose an apology email to my boss."
"Aide moi àécrire une lettre àmon père."

- write_fiction:
"Write a poem about the sunset."
"Create a short story about a time-traveling astronaut."
"Make a rap in the style of Drake about the ocean."
"Escribe un cuento sobre un niño que descubre un tesoro,
      pero después viene un pirata."
"Compose a sonnet about time."

- how_to_advice:
"How do I turn off my screensaver?"
"My car won\'t start; what should I try?"
"Comment faire pour me connecter àmon wifi?"
"What\'s the best way to clean hardwood floors?"
"How can I replace a flat tire?"

- creative_ideation:
"What should I talk about on my future podcast episodes?"
"Give me some themes for a photography project."
"Necesito ideas para un regalo de aniversario."
"Brainstorm names for a new coffee shop."
"What are some unique app ideas for startups?"

- tutoring_or_teaching:
"How do black holes work?"
"Can you explain derivatives and integrals?"
```

```
"No entiendo la diferencia entre ser y estar."
"Explain the causes of the French Revolution."
"What is the significance of the Pythagorean theorem?"

- translation:
"How do you say Happy Birthday in Hindi?"
"Traduis Je t\'aime en anglais."
"What\'s Good morning in Japanese?"
"Translate I love coding to German."
"¿Cómo se dice Thank you en francés?"

- mathematical_calculation:
"What is 400000 divided by 23?"
"Calculate the square root of 144."
"Solve for x in the equation 2x + 5 = 15."
"What\'s the integral of sin(x)?"
"Convert 150 kilometers to miles."

- computer_programming:
"How to group by and filter for biggest groups in SQL."
"I\'m getting a TypeError in JavaScript when I try to call
      this function."
"Write a function to retrieve the first and last value of
      an array in Python."
"Escribe un programa en Python que cuente las palabras en
      un texto."
"Explain how inheritance works in Java."

- purchasable_products:
"iPhone 15."
"What\'s the best streaming service?"
"How much are Nikes?"
"Cuánto cuesta un Google Pixel?"
"Recommend a good laptop under $1000."

- cooking_and_recipes:
"How to cook salmon."
"Recipe for lasagna."
"Is turkey bacon halal?"
"Comment faire des crêpes?"
"Give me a step-by-step guide to make sushi."

- health_fitness_beauty_or_self_care:
"How to do my eyebrows."
"Quiero perder peso, ¿cómo empiezo?"
"What\'s a good skincare routine for oily skin?"
"How can I improve my cardio fitness?"
"Give me tips for reducing stress."
```

```
- specific_info:
"What is regenerative agriculture?"
"What\'s the name of the song that has the lyrics I was
      born to run?"
"Tell me about Marie Curie and her main contributions to
      science."
"What conflicts are happening in the Middle East right
      now?"
"Quelles équipes sont en finale de la ligue des champions
      ce mois-ci?"
"Tell me about recent breakthroughs in cancer research."

- greetings_and_chitchat:
"Ciao!"
"Hola."
"I had an awesome day today; how was yours?"
"What\'s your favorite animal?"
"Do you like ice cream?"

- relationships_and_personal_reflection:
"What should I do for my 10th anniversary?"
"I\'m feeling worried."
"My wife is mad at me, and I don\'t know what to do."
"I\'m so happy about my promotion!"
"Je sais pas ce que je fais pour que les gens me
      détestent. Qu\'est-ce que je fais mal?"

- games_and_role_play:
"You are a Klingon. Let\'s discuss the pros and cons of
      working with humans."
"I\'ll say a word, and then you say the opposite of that
      word!"
"You\'re the dungeon master; tell us about the mysterious
      cavern we encountered."
"I want you to be my AI girlfriend."
"Faisons semblant que nous sommes des astronautes. Comment
      on fait pour atterrir sur Mars?"

- asking_about_the_model:
"Who made you?"
"What do you know?"
"How many languages do you speak?"
"Are you an AI or a human?"
"As-tu des sentiments?"

- create_an_image:
"Draw an astronaut riding a unicorn."
"Photorealistic image of a sunset over the mountains."
"Quiero que hagas un dibujo de un conejo con una corbata."
"Generate an image of a futuristic cityscape."
"Make an illustration of a space shuttle launch."

- analyze_an_image:
"Who is in this photo?"
"What does this sign say?"
"Soy ciega, ¿puedes describirme esta foto?"
"Interpret the data shown in this chart."
"Describe the facial expressions in this photo."

- generate_or_retrieve_other_media:
"Make a YouTube video about goal kicks."
"Write PPT slides for a tax law conference."
"Create a spreadsheet for mortgage payments."
"Find me a podcast about ancient history."
"Busca un video que explique la teoría de la relatividad."

- data_analysis:
"Here\'s a spreadsheet with my expenses; tell me how much
      I spent on which categories."
"What\'s the mean, median, and mode of this dataset?"
"Create a CSV with the top 10 most populated countries and
      their populations over time. Give me the mean annual
      growth rate for each country."
"Perform a regression analysis on this data."
"Analyse these survey results and summarize the key
      findings."

- unclear:
"If there is no indication of what the user wants; usually
      this would be a very short prompt."

- other:
"If there is a capability requested but none of the above
      apply; should be pretty rare."
```

```
Okay, now your turn, taking the user conversation at the
      top into account: What capability are they seeking?
      (JUST SAY A SINGLE CATEGORY FROM THE LIST, NOTHING
      ELSE). If the conversation has multiple distinct
      capabilities, choose the one that is the most
      relevant to the LAST message in the conversation.
      The conversation to classify
      is:\n{conversation_text}.
```

Additional Results on Topic Discovery. We provide detailed statistics across 5 platforms in the Figure 9.

A.6 Conversation Completeness Analysis

The pipeline implements a three-stage workflow adapted from the DeepEval[4] conversation completeness metric framework. First, we extract user intentions from each conversation by prompting Qwen3-8B (Yang et al., 2025) with a temperature of 0.7 to identify the distinct goals or information needs expressed in the user turns, yielding a chronologically ordered list of intentions per conversation. Second, for each identified intention, we construct an evaluation prompt that presents the full conversation context alongside the specific intention and instruct Qwen3-8B to classify whether that intention was satisfied; the model produces a categorical verdict of *complete* (the assistant fully addressed the request), *partial* (the assistant provided some relevant information but the goal was incompletely met), or *incomplete* (the assistant failed to address the request or the conversation ended before meaningful progress). Finally, we aggregate these intention-level verdicts into a conversation-level completeness score by computing the weighted proportion of intentions, where complete verdicts contribute 1.0, partial verdicts contribute 0.5, and incomplete verdicts contribute 0.0. This three-way categorical framework allows us to capture nuanced degrees of conversational success beyond binary outcomes. We apply this pipeline uniformly across all five platforms in SHARECHAT, enabling direct cross-platform comparisons of how effectively different assistants satisfy user needs over the course of extended interactions. During manual coding, we observe that mis-classification often occurred when users pasted text, such as articles or messages, without explicit instructions or questions. In these instances, the LLM generate responses based solely on an inferred understanding of the content.

Figure 10 shows the conversation-level completeness scores, where ChatGPT, Claude, Grok,

[4] https://github.com/confident-ai/deepeval?tab=readme-ov-file

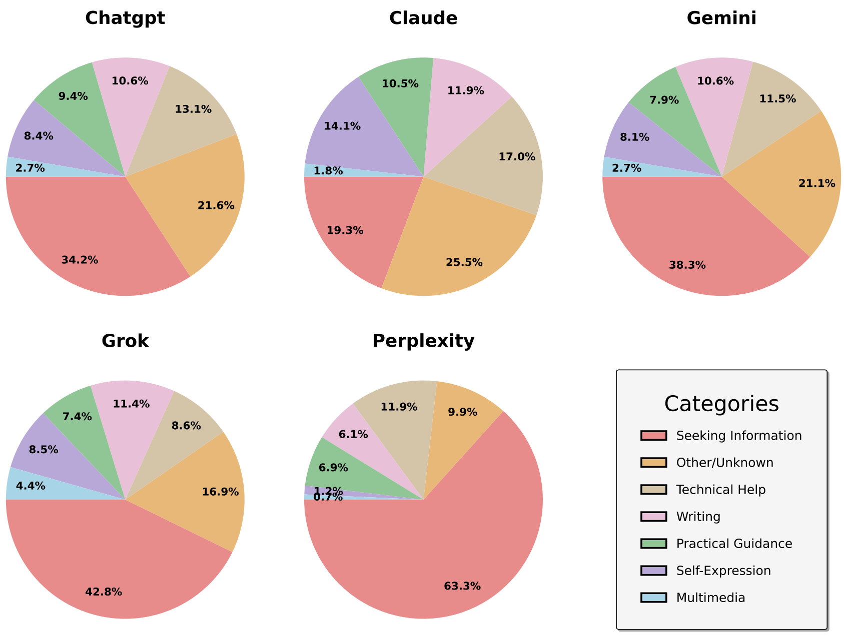


Figure 9: Topic distribution of user requests across five platforms.

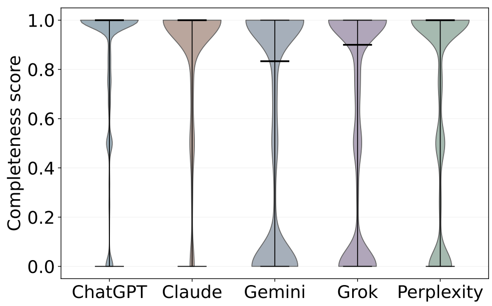


Figure 10: Completeness score distribution

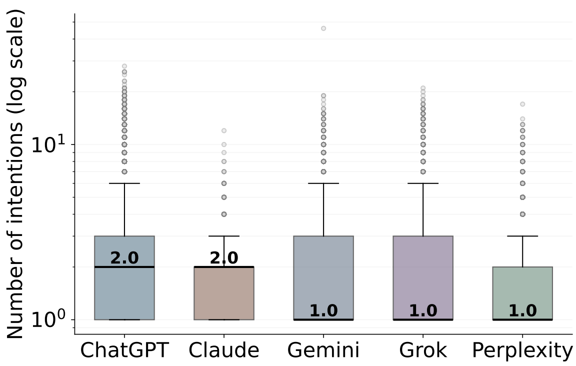


Figure 11: Number of intentions per conversation. It presents the distribution of the number of extracted intentions per conversation on a logarithmic scale, with median values labeled above each box.

and Perplexity all achieve median scores of 1.0, while Gemini's median is 0.833. Figure 11 shows the number of interactions across platforms.

The prompts used are as follows. To handle conversations that exceed the 40,960-token context limit of Qwen3-8B, we apply token-based truncation to assistant responses while preserving all user content, and we exclude conversations requiring severe truncation (retaining less than 80% of content) to maintain evaluation quality.

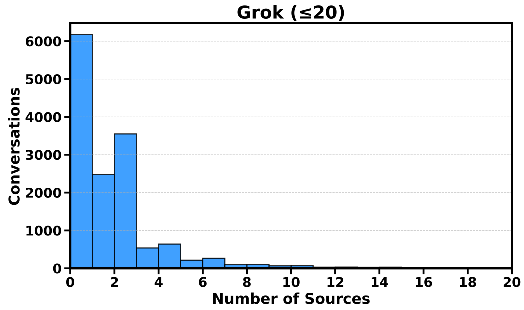
Prompt A: User Intention Extraction

```
### SYSTEM ROLE
You are an expert Conversation Analyst.

### TASK
Extract a chronological list of **User Intentions** from
      the conversation log.

### INSTRUCTIONS
1. **Identify Distinct Goals:** Focus on information
      seeking, task requests, or problem-solving goals.
2. **Maintain Order:** The first item in your list must
      correspond to the user's first real request, and so
      on.
3. **Ignore Noise:** Skip purely social turns (e.g.,
      "Hello", "Thank you", "Okay") unless they are the
      only message.

### OUTPUT FORMAT
Respond with a raw JSON object enclosed strictly within
      <output> tags.
The JSON must have exactly one field: "intentions" (a list
      of strings).

### EXAMPLE

Input Turns:
[
    {"role": "user", "content": "Hi, I need help with
          Python."},
    {"role": "user", "content": "How do I reverse a
          list?"},
    {"role": "user", "content": "Thanks. Also, what is the
          weather in Tokyo?"}
]
```

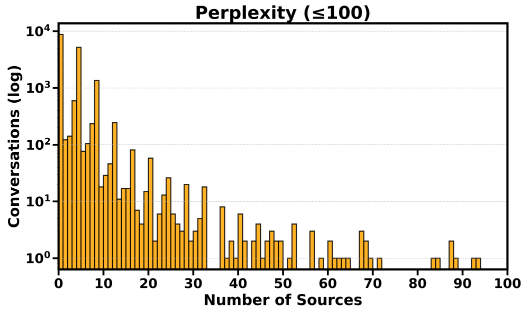
```
Correct Output:
<output>
{
    "intentions": [
        "User wants to know how to reverse a list in
              Python",
        "User wants to check the weather in Tokyo"
    ]
}
</output>

### CURRENT INPUT
Turns:
{{CONVERSATION_TEXT}}

### RESPONSE
Generate the JSON response now.
1. Start your response with the opening tag <output>.
2. Ensure the JSON is valid.
3. End with the closing tag </output>.
```

Prompt B: Conversation Completeness Labeling

```
### SYSTEM ROLE
You are an expert Quality Assurance Evaluator for AI
       conversations.

### TASK
Determine if the specific **User Intention** was satisfied
        by the LLM based on the conversation history.

### CRITERIA
- **Verdict: "yes"** if:
    1. The LLM provided the correct information, code, or
          creative output requested.
    2. The user explicitly expressed satisfaction (e.g.,
          "Thanks", "That works").
    3. The interaction reached a logical conclusion where
          the goal was met.

- **Verdict: "partial"** if:
    1. The LLM started addressing the request but the
          conversation ended before completion.
    2. The LLM provided some relevant information but
          missed key aspects of the request.
    3. The LLM gave a partial solution that requires
          additional steps the user would need to complete.
    4. The user asked follow-up questions indicating
          partial understanding/satisfaction.

- **Verdict: "no"** if:
    1. The LLM refused the request (unless it was a safety
          violation).
    2. The LLM completely misunderstood the request or
          provided irrelevant information.
    3. The user expressed frustration or repeatedly asked
          the same thing without progress.
    4. The LLM asked for clarification but the
          conversation ended before any attempt to help.

### OUTPUT FORMAT
Respond with a raw JSON object enclosed strictly within
       <output> tags.
The JSON must have these fields:
- "intention": (repeat the intention text)
- "verdict": (value must be "yes", "partial", or "no")

### EXAMPLE

Intention: "User wants to learn about Python decorators"
Turns: [
    {"role": "user", "content": "Can you explain Python
          decorators?"},
    {"role": "assistant", "content": "Sure! Decorators are
          functions that modify other functions. They use @
          syntax. Would you like to see a code example?"}
]

Correct Output:
<output>
{"intention": "User wants to learn about Python
       decorators", "verdict": "yes"}
</output>

### CURRENT INPUT
Intention: {{USER_INTENTION}}

Turns:
{{CONVERSATION_TEXT}}

### RESPONSE
Generate the JSON response now.
1. Start your response with the opening tag <output>.
2. Ensure the JSON is valid.
3. End with the closing tag </output>.
```



(a) Grok



(b) Perplexity

Figure 12: Distribution of source citations per conversation for Grok and Perplexity. Extreme tails are omitted for visual clarity; the maximum observed source count is 83 for Grok and 1,059 for Perplexity.

A.7 Additional Response Source Analysis: Count Distribution

To understand the retrieval intensity of search-enabled platforms, we analyze the distribution of source counts per conversation. Figure 12 shows that Grok typically uses very few sources per conversation, whereas Perplexity exhibits a long-tailed distribution in which many conversations draw on dozens of sources.

A.8 Additional Temporal Analysis: Usage Rhythms and Latency

We expand on the temporal dynamics of the dataset by examining both the diurnal rhythms of user activity and the granular relationship between content length and reading time. First, to understand

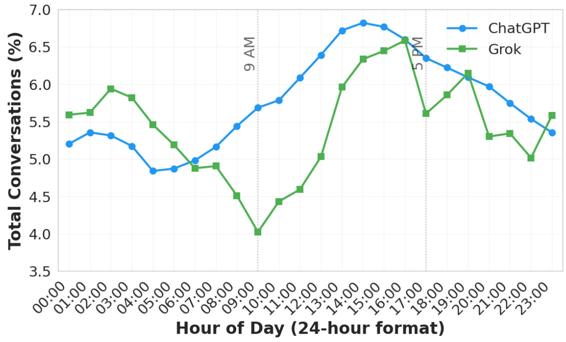
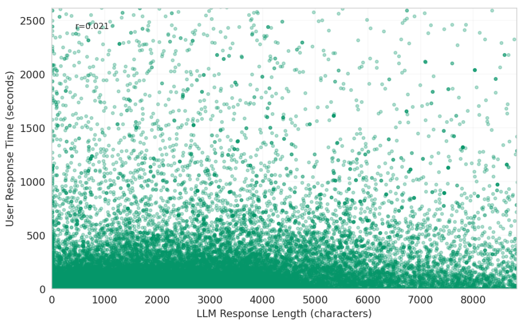
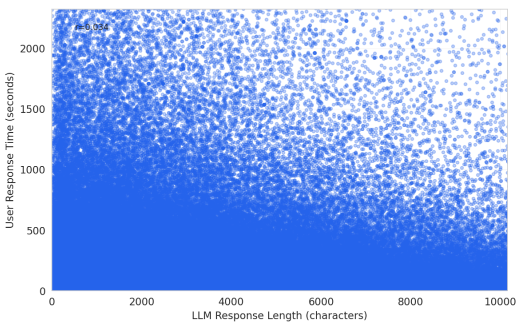


Figure 13: Comparison of temporal activity displaying hourly activity.

when users are most active, Figure 13 presents the normalized hourly activity for each platform, with timestamps adjusted to the users' local time. Second, we investigate whether longer model outputs consistently drive longer user engagement. Figure 14 plots the raw distribution of user response times against LLM response length. The scatter plot reveals a high degree of variance, with Pearson correlations near zero for both platforms (ChatGPT: $r = 0.034$; Grok: $r = 0.021$). This indicates that while aggregate trends exist, model verbosity is a poor predictor of dwell time.



(a) Grok: LLM response length vs. user response time



(b) ChatGPT: LLM response length vs. user response time

Figure 14: Analysis of the relationship between LLM response length and user response time.