Show or Tell? Modeling the evolution of request-making in Human-LLM conversations

Anonymous ACL submission

Abstract

Chat logs provide a rich source of information about LLM users, but patterns of user behavior are often masked by the variability of queries. We present a new task, segmenting chat queries into contents of requests, roles, query-specific context, and additional expressions. We find that, despite the familiarity of chat-based interaction, request-making in LLM queries remains significantly different from comparable human-human interactions. With the data resource, we introduce an important perspective of diachronic analyses with user expressions. We find that query patterns vary between early ones emphasizing requests, and individual users explore patterns but tend to converge with experience. Finally, we show that model capabilities affect user behavior, particularly with the introduction of new models, which are traceable at the community level.

1 Introduction

Chat-based interfaces with underlying LLMs have created an unprecedented paradigm for AI (Zhu et al., 2025; Gao et al., 2024). The versatile conversation format enables open-ended, customized user inputs, creating a new form of interaction. Meanwhile, these dynamics present interesting challenges. How do users learn and adapt their behaviors and expectations to converse with the system (Choi et al., 2024; Schroeder et al., 2024)? Do users apply similar expressions in such interactions to natural conversations (Nass and Moon, 2000)? And, how do their utterances change when the model itself is updated (Ma et al., 2024)?

Nonetheless, very limited attention has been given to these *language* aspects, much less the *evolution* of use patterns in the textual expressions across time. More often, studies of real-world chat logs focus on the semantics and tasks conveyed (Tamkin et al., 2024; Cheng et al., 2025; Handa et al., 2025), whereas users' expressions, a more fundamental and generalized linguistic feature of user-LLM interaction, remain understudied.

In this work, we take first steps to analyze how these linguistic inquiries are formed, with a user's intent and the information they provide. and provide detailed analysis of patterns in interactions over relative and absolute time. First, we introduce a new task of request segmentation to tackle the challenge of users freely embedding *requests* in their *expressions*, together with other parts with substantial presence like *contexts* and assigned *roles*. Via an extendable, semi-automatic LLM annotation workflow, we present a dataset on top of WildChat (Zhao et al., 2024) with 211,414 parsed user utterances consisting Request Content, Context, Roles, and Expressions (ReCCRE). The dataset features clean separation of the content and context specific to a request from the generic natural language expressions used to deliver the request.

Using this dataset, we make several key observations and conclusions. We show that the LLM chat modality is fundamentally different from similar natural request-making conversations between humans, by comparing with the Stanford Politeness Datasets (Danescu-Niculescu-Mizil et al., 2013), a primary resource in computational pragmatics. We identify key repeating patterns in queries, which range on an axis between *request-centric* and *context-infused*. More importantly, we introduce how the perspective of expressions enables *diachronic* user modeling, with use records as a time-lapse data source. We discuss how the dimension of time provides new angles in understanding user lifecycles and the collective user base. We find clear traces that, as users gain familiarity with the system, they tend to change expression formats less, and they migrate from simple chunks of requests to more context-heavy combinations. Finally, we exhibit the possibilty of full-scale community analysis and discuss patterns, e.g., the impact of the introduction of new models.

2 Related Work

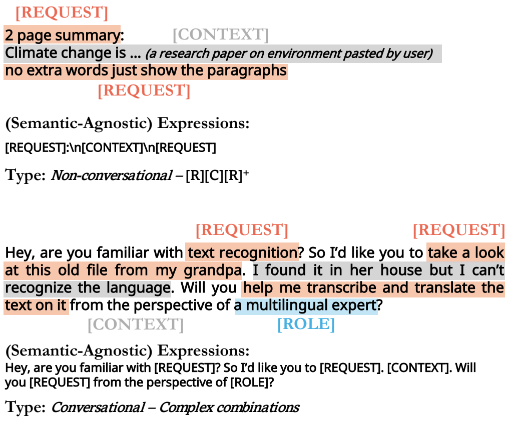
Interpreting Real-World Human-LLM Conversations has been a rising topic thanks to new data resources (Zhao et al., 2024; Zheng et al., 2024a). However, the major body of work has focused on the detection and categorization of what tasks users use LLMs for (Zhang et al., 2025b; Cheng et al., 2025; Mireshghallah et al., 2024) and their impacts (Handa et al., 2025; Tamkin et al., 2024; Kirk et al., 2024), as well as specific features such as values (Huang et al., 2025) and jailbreaking attempts (Jin et al., 2025). Beyond NLP, the formatting of prompts as well as the broader user experience with LLM input interfaces has also been core topics in Human-Computer Interaction (Zamfirescu-Pereira et al., 2023; He et al., 2025; Gao et al., 2024; Zhang et al., 2025a).

Some existing work shares the focus on individual attributes relevant to our discussions. For instance, Huang et al. (2024) concerns the "conversational tones" shared or (mis)aligned between human and LLMs, and Zheng et al. (2024b) probes LLM performance with different role assignments. However, our work is fundamentally different: Prior work usually targets what a human-LLM conversation *should* look like, a dominant thread related to fine-tuning and deployment (Mott et al., 2024; Mishra et al., 2022; Ivey et al., 2024); whereas our work focuses on the mining of existing data and new analysis paradigms. More importantly, we present a new systematic view that is not covered by the studies on single attributes.

At high level, our work is most related to Mysore et al. (2025) and Kolawole et al. (2025), which also seek to extract and describe latent patterns from massive inputs. However, both works have distinct goals: the former marks "PATHs" as a dialog proceeds and focuses on assisted writing, and the latter targets task taxonomies with semantic similarities.

3 Annotating User Request-Making

User requests are complicated and have many parts, some of which are more or less responsive to user choices. For example, a user seeking to generate a document may start with an external information need, such as the topic of an email or an essay question. Other parts of the request are more subject to user choices: a user may assign the LLM a role, specify restrictions on output, or add additional social cues.



2 page summary:
Climate change is ... *(a research paper on environment pasted by user)*
no extra words just show the paragraphs

(Semantic-Agnostic) Expressions:

[REQUEST]:\n[CONTEXT]\n[REQUEST]

Type: *Non-conversational* – [R][C][R]$^+$

Hey, are you familiar with text recognition? So I'd like you to take a look at this old file from my grandpa. I found it in her house but I can't recognize the language. Will you help me transcribe and translate the text on it from the perspective of a multilingual expert?

(Semantic-Agnostic) Expressions:
Hey, are you familiar with [REQUEST]? So I'd like you to [REQUEST]. [CONTEXT]. Will you [REQUEST] from the perspective of [ROLE]?

Type: *Conversational – Complex combinations*

Figure 1: Two examples of user input annotated with request content, context, and roles. The precise definitions of the components are discussed in §3.1, and the expression types are elaborated in Table 1.

Compared with *what* LLMs are used for, *how* the conversational user-LLM interaction happens remains largely understudied in the field, especially from the pragmatics aspect. When studying user behavior, we may want to ignore the externally defined context and focus on those other aspects. We start by constructing the data infrastructure for modeling user request-making behaviors by segmenting requests. Our goal is a generalizable corpus annotation scheme that enables systematic analysis and comparison of user requests, and allows adapation to unseen data and natural language conversations to support comparisons.

3.1 Task Setup

Source Data We collect 317,373 initial user inputs (i.e., first turns) from the WildChat dataset (Zhao et al., 2024) as the base corpus.

The Annotation Task As the first step, we need to distinguish between the making of an effortful request and the direct retrieval of answers (e.g., "who is the 3rd president of the U.S."). While the latter represents another common type of usage, the engagement level is low; it is less likely to involve a conversational scenario, and the dynamics are remarkably different from request-making. In practice, the annotator first reads the input text thoroughly and determines whether it involves a request or a direct question, or "both" or "neither".

Next, for the request-making cases, we outline the case-specific core semantics relevant to the request. This therefore filters out the case-independent framing templates used to deliver the request, which would enable comparison across

Type		Description	Examples
Non-conversational	$[R]$	A single *Request* component.	$[R]$ Give me $[R]$.
	$[R] * n$	Multiple *Request*s concatenated in simple ways, without conversational expressions.	$[R]$ and $[R]$. $[R]$. $[R]$. Also $[R]$.
	$[R][C]$	One *Request* followed by One *Context* component, without conversational expressions.	$[R]$: $[C]$ $[R]$ such as $[C]$.
	$[C][R]$	One *Context* followed by One *Request* component, without conversational expressions.	$[C]$. Now, $[R]$. $[C]$\n\n$[R]$
	$[R][C][R]^+$	A pair of *Request* and *Context*, followed by additional *Request* components.	$[R]$, $[C]$. $[R]$ and $[R]$. $[R]$: $[C]$. $[R]$. $[R]$. $[R]$.
	$[R][C][C]^+$	A pair of *Request* and *Context*, followed by additional *Context* components.	$[R]$ based of $[C]$: $[C]$. $[R]$. $[C]$. $[C]$. $[C]$.
	$[C]^+$	Concatenation of *Context* components only. Usually seen in early requests to complete writings.	$[C]$. $[C]$ $[C]$
	Other $[R]/[C]/[role]$ compositions	Other more complicated series of $[R]$, $[C]$, and $[role]$, with simple, non-conversational expressions.	$[R]$, $[C]$. Then, $[R]$, $[C]$. Finally, $[R]$, $[C]$. $[C]$. $[C]$. Given $[C]$, $[R]$.
Conversational	Single $[R]$	One single *Request* in a conversational expression.	Can you help me to $[R]$? Hi I wanna $[R]$.
	Simple $[R]/[C]/[role]$ combinations	Simple combinations of *Request*, *Context*, and *Role* using conversational expressions.	You are $[role]$. Now, $[R]$. I'm working on $[C]$ and I'd like you to $[R]$.
	Complex compositions	Other more complicated series of $[R]$, $[C]$, and $[role]$, with full, conversational expressions.	How can I $[R]$? $[R]$. Please be sure to $[R]$! Act as $[role]$ and $[R]$. You will $[R]$: $[C]$.

Table 1: Taxonomy of user expressions, with 8 non-conversational types and 3 conversational types.

different dialogs. Specifically, we introduce the following annotations of three elements of requests plus the user expressions, on top of the full user input text:

- **Request Content** (**$[R]$**): A core span of text that specifies what task(s) exactly the user wants the LLM to perform, or what goal(s) the user wants to achieve;

- **Context** (**$[C]$**): A detailed span of context information that does not directly constitute the request, but provides support for neighboring requests. This includes the chunks of "target text" that the LLMs are requested to process (e.g., the pasted article for the request "2 page summary" in Fig. 1).

- **Role** (**$[role]$**): Any roles that the LLM is asked to take on to achieve the requests.

- **Expression**: The remaining text after extracting the above components, which represent the generic language templates used to embed and deliver the requests.

Each word in a request is assigned to precisely one of the four categories, and the labels thus form a non-overlapping full division of the user input. This forms the basis of the corpus annotation task, and we move on to discuss the implementation on the large-scale data.

3.2 Automating the annotation pipeline

To adapt the annotation to chat logs at the million-request scale, we seek to balance between automation and reliability. As the annotation of request segments is highly context-dependent and may involve intrinsic ambiguity, we implement a review-and-revise pipeline that simultaneously generates semi-supervised annotations and extends the available data for fine-tuning with consistent standards learned from human annotation.

The pipeline involves three contributors: a human annotator; an interim SotA LLM specialized in text understanding (L); and a smaller local LLM as full-scale annotator (l). We bootstrap from a small number of hand-labelled data, use L as pseudo-reference for fine-tuning l, and eventually automate annotations with the fine-tuned local l.

First, the human annotator manually labeled a small, random batch of "root" data. This small collection of references is consistent and we denote this *gold* dataset as D_0. Both LLM annotators are then provided with a detailed prompt of annotation rules[1] and fine-tuned based on D_0. Next, both L and l are evaluated on a separate, randomly sampled set D_1^{raw}. We then convert this raw subset into a *silver* set D_1 as follows:

- If the two models *agree* on the instance (the total of different labels by L and l is lower than threshold δ), the annotation of L is accepted and added to D_1.

- Otherwise, if the two models *disagree*, the instance is manually reviewed. The annotator determines which of L or l is acceptable, following the standards of D_0. If both are incorrect, they manually label the instance. The reviewed/relabelled version is added to D_1.

[1] The prompts are available in Appendix B.

D_1 as a silver set is then merged with D_0 to form an expanded annotated dataset for fine-tuning. Both L and l are then fine-tuned and evaluated on another new batch D_2^{raw}, and the process is repeated so on and so forth. Finally, after fine-tuning the models with incremental, semi-supervised data, we migrate the model prediction process from the black-box, high-cost L to our local model l to perform full-scale automated annotation.

We use `gpt-4o-2024-08-06` as the intermediate L, and a flagship 10B-level open-source LLM at the time of the work, `Qwen2.5-14B-Instruct`, as the full-scale annotator l. The review-and-revise loop was repeated 3 times on a total of 762 instances, and a template-based verification showed that ill-formatted responses are $< 0.5\%$.

4 The ReCCRE dataset

4.1 Basic information

After obtaining the valid annotations, we set up a spam filter to remove consecutive dialogs that are overly similar or created within a very short period, and also remove the instances that do not involve request-making. This results in a collection of 211,414 user request-making inputs from 18,964 users. The dataset covers the time window from April 2023 to May 2024, and spans 6 versions of the `gpt-3.5-turbo` and `gpt-4` models.[2]

Long-term Users To track the change of use patterns over time, we focus on the core users with sufficient experience and time to play with and adapt to the system. In practice, we seek users (1) whose use records (time between first and last dialog created) span more than 14 days, and (2) have started at least 10 dialogs. This yields a subset of 2,092 users with 59,175 request-making user inputs in total. We refer to this key group as *long-term* or *stable* users. Our analyses will primarily focus on this group, as it more reliably reflects use patterns and provides the legitimacy for diachronic observations. We present a full-scale case study comparing long-term users with all users and the full data in §5.2 under the Lexical Richness framework.

4.2 Human-LLM requests differ from human-human requests

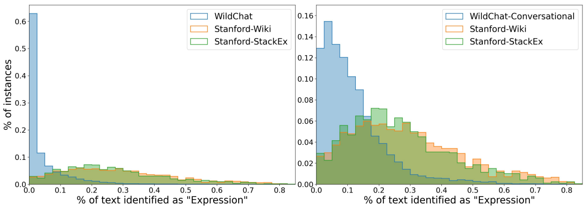


Figure 2: The relative amount of formatting in Wildchat user inputs (left) and the conversational portion only (right), compared with the Stanford Politeness Dataset.

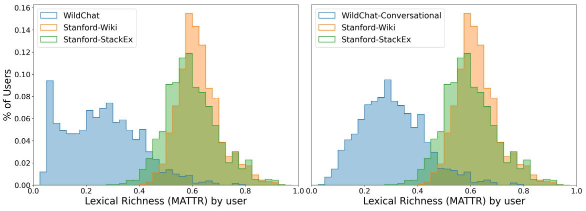


Figure 3: Distribution of speaker Lexical Richness (measured by Moving-Average Text-Token Ratio) in Wildchat (left) and its conversational portion only (right), compared with the Stanford Politeness Dataset.

Factoring out external context and focusing on request expression allows us to compare recorded LLM chats to human-human interaction. To evaluate this difference, we applied the same request segmentation code to the Stanford Politeness Datasets (Danescu-Niculescu-Mizil et al., 2013), a widely used pragmatics corpus relevant to request-making. The corpus involves two datasets collected from natural dialog sources, the Wikipedia editor discussions (Stanford-Wiki) and the Stack-Exchange Q&A forum (Stanford-StackEx). Each utterance is designed to involve the speaker making requests to another member. We can therefore compare how requests are delivered in the two human-human dialog scenarios and the user-LLM scenario.

We note the distinct composition of the utterances and the Lexical Richness of expressions (Laufer and Nation, 1995; Shen, 2022). Figures 2 and 3 respectively compares (1) the percentage of input text as expression and (2) the user-level Moving-Average Text-Token Ratio (Covington and McFall, 2010) of ReCCRE and Stanford datasets. The two natural subsets share highly similar stats despite distinct sources and contexts; however, ReCCRE from WildChat uses qualitatively fewer expressions to embed requests, and user language has much lower diversity. One major reason is the presence of non-conversational "imperative" spans (Table 1); for instance, "Write an article about jogging." is marked as a single $[R]$ component with minimal or no formatting. This is a common paradigm, different from natural con-

[2] See Zhao et al. (2024) for the detailed documentation.

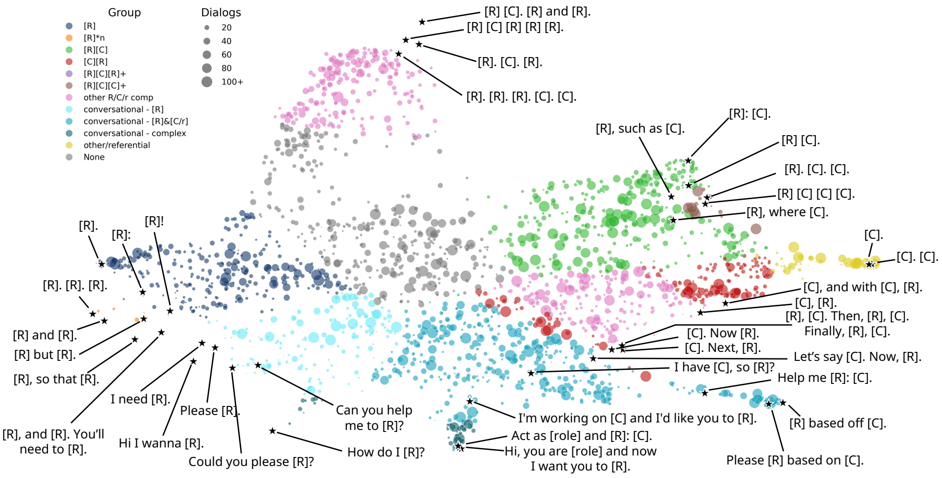


Figure 4: The overview of the ReCCRE dataset as a user-level 2-D plot. Each circle represents a user, with its size matching their total dialogs and color clustered base on closest anchor point. In general, the horizontal axis displays the ratio of $[R]$ an $[C]$, and the vertical axis ranges from the most to least conversational.

versations, as the latter would require more appropriate social grounding. However, if we confine to the *conversational* instances (the subplots on the right in both figures) by removing the simple combinations of request content and context, we see that the difference still holds. In other words, the existence of the non-conversational imperatives is not sufficient to account for the difference; there exist fundamental contrasts of langauge use independent from request content that are specific to the human-LLM scenario.

4.3 Categorization and Illustration

4.3.1 Taxonomy of Expressions

To understand and generalize the annotations, we start from a taxonomy of expressions in request-making shown in Table 1. We refer to an expression as *conversational* if it involves explicit signs of conversing with another party, such as person ("You" and "I"), politeness strategies, and greetings. Conversely, if an utterance is a mere imperative combination of $[R]$ and $[C]$ without signs of conversation, it is marked as *non-conversational*. We further break down the expressions based on their structure and complexity, e.g., whether multiple request or context components are involved.

Anchor Points To calibrate the varied user inputs, we collect 40 different instances of the most frequent expressions in the dataset, covering all 11 categories. We use them as *anchor points* in our analyses (including subsequent figures) to provide reference for visualizations, and more importantly, to allow categorization of arbitrary unseen expressions based on their closest anchor points.

4.3.2 Visualizing the data distribution

We vectorize user expressions with a SotA text-embedding model, `gte-large-en-v1.5` (Li et al., 2023), with $[R]$, $[C]$, and $[role]$ wrapped as formatted placeholder tokens (`__[REQUEST]__`, etc.) Each request-making utterance is encoded as a 1024-dim vector, and a user is represented by the average of their utterances. We then map all users as well as the anchor points to a 2-D space using PaCMAP (Wang et al., 2021), a SotA dimension reduction (DR) method. In this way, the collections of user expressions are depicted as a scatter plot in Figure 4, where each bubble represents a long-term user. The bubble size is in proportion to the number of dialogs a user created, and they are categorized and colored based on the type of the closest anchor point. If a user representation is not close enough to any anchor points, it is marked in grey.

Note that the horizontal layout corresponds to the composition of Goals ($[R]$) and Context ($[C]$), where the leftmost represents the sole $[R]$ and the rightmost corresponds to $[C]$ only, and the combined forms are in between. Meanwhile, the vertical positions can be interpreted as the "conversationality": the bottommost ones, featuring role assigning and politeness patterns like "please", are closest to the expressions and force of natural conversations; the topmost ones, indicating a repetitive sequence of $[R]$ and $[C]$ with no additional text, is most tool-like and unlikely in human-human request-making.

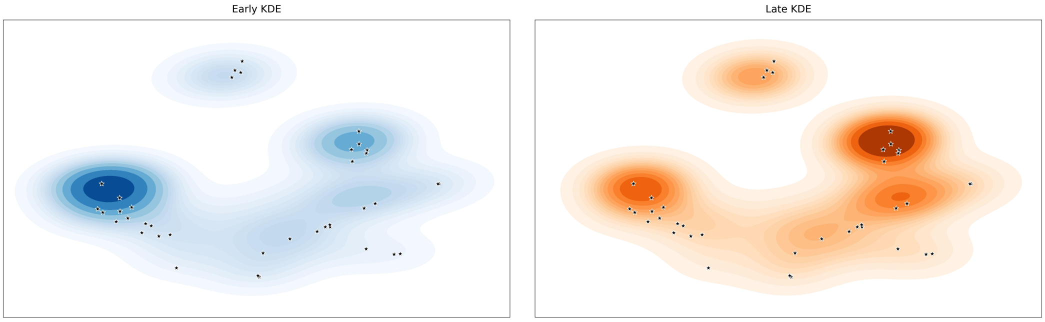


Figure 5: Distribution of the 1st dialogs (left) and 20th dialogs (right) of eligible long-term users as Kernel Density Estimation (KDE) plots in the same 2D space from Fig. 4.

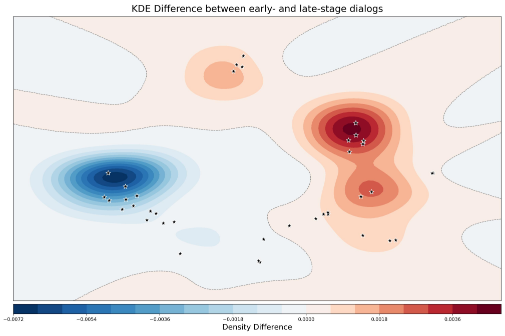


Figure 6: Difference of densities between the early and late inputs of the long-term users (Fig. 5). In the later stage, the balanced combination of $[R]$ and $[C]$ (upper-right area in red) sees major inflows, while the type of non-conversational stacks of $[R]$s (blue area on the left) significantly decreases.

5 Diachronic analysis of Request-making in the ReCCRE dataset

Enabled by the new data infrastructure, we move on to discuss a novel paradigm: modeling user behaviors and patterns in a *diachronic* manner, thereby understanding interaction as a systematic and dynamic process. We will first inspect the lifecycles of individual users, and discuss how the fundamental properties, such as the diversity of expression types, change across time. Next, we interpret the holistic evolution patterns formed by the community collectively, and extend further to compare the core user base and the lay public.

5.1 Modeling User Lifecycle

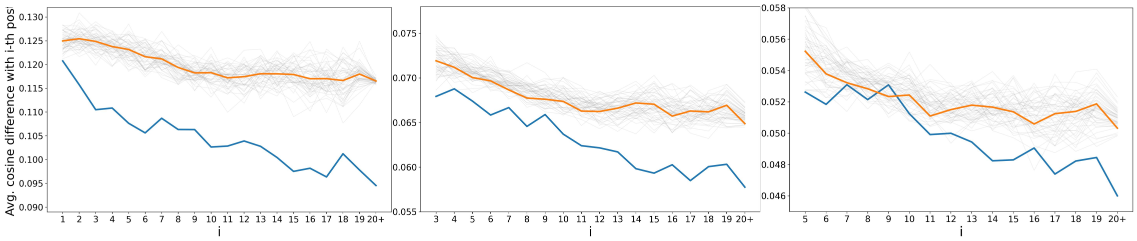
Long-term, non-intrusive documentary of user-LLM interactions (Zhu et al., 2025) provides an interface for the user lifecycles, i.e., the full course of actions and use history. Here, we discuss how a fully text-based analysis can help to model the change of use patterns across time.

5.1.1 What expressions were used and how are they structured?

As a natural continuation of Figure 4, we further add the dimension of time and inspect the most common patterns across myriad individual dialogs at different stages. To model the user lifecycles, we position a pair of early- and late-stage input data from long-term users in the same 2-D space from Fig. 4 and apply Kernel Density Estimation to model the frequency of user expressions. Figure 5 shows the comparison of the 1st (left) and the 20th (right) request-making dialog of all eligible long-term users, with the same anchor points from Fig. 4, and we directly depict the difference between the two KDEs in Figure 6. For the users' first explorations, most utterances are made up of only one $[R]$ or the simple combination of two. This describes the exploratory stage of interaction with the system with more concise and generic requests. However, as users gain familiarity, this pattern drastly decreases (though still a major type), and the mass is significantly transitioned to the addition of more specific contexts on the right side, as well as the more complex type of multiple goals and context (top). This suggests a communal shift towards requests with higher specificity and complexity: Users are accustomed to tailoring requests with more context details and fine-grained content later on. This "show or tell" transition reemphasizes that, whereas users may develop distinct use scenarios and tasks, there are indeed fundamental commonalities to be mined at the expression level.

5.1.2 User-level Evolutions

The user expression space like Fig. 4 enables the study focusing on individual users. For an intuitive



(a) W=1

(b) W=3

(c) w=5

Figure 7: Convergence of user expressions over time.

instance, we can visualize individual trajectories as their latest inputs move along the 2-D space; this is exemplified by the distinct but discernible trajectories of 6 random long-term users illustrated in Figure 9 in Appendix A.

Further, we also wonder if these paths of users can be collected across users to suggest deeper trends. As a representative example, we consider the effect of familiarity on a user's tendency to repeat same kinds of expressions they have used previously. One hypothesis is that a user's expressions may converge with more experience, as they develop their "go-to" choices and stick to how that have worked in previous cases. Alternatively, as users gain familiarity, they may understand the capabilities and boundaries of the LLMs better, and thus rely less on rigorous prompt formats and interact in more casual, arbitrary ways.

To test the contrasting diachronic hypotheses, we examine the minimal difference between a long-term user's input and their most recent inputs, i.e., between their i-th and its previous k request utterances, $\min_{j=1,\ldots,k-1}[1 - sim(U_i, U_{i-j})]$, for all valid i given window size k. Then, we collect the minimal difference across all users and compute the average for each position i, to illustrate the averaged step-by-step convergence (or divergence) of expressions within a long-term user's lifecycle.

Figure 7 displays the results under different window sizes k. The actual chronological data is shown as the blue line. It is compared against the average of 50 random trials where the dialogs from the same user are randomly shuffled (shown in orange), thus breaking the diachronic ties. We observe strong evidence for the *convergence* hypothesis across time: the difference between a new input and its immediate predecessors sees a drastic, continuous decline as users continue to interact with the system. This is quantitatively different from the baseline level of random shuffles, and the difference between the two gets more significant with more input requests. This suggests that users overall develop rather stable patterns of usage as they gain familiarity. Meanwhile, we also note that this pattern is most significant with window size $k = 1$ (Fig. 7a), i.e., when comparing each input with the exact one previous dialog. The gap between real and random situations is reduced with $k = 3$ (Fig. 7b) and further with $k = 5$ (Fig. 7c). This indicates that the most recent cases consistently serve as more significant references for a new user request, whereas the effect of earlier inputs decreases with their lower recency.

5.2 Picturing the community

So far we have modeled the evolving request-making expressions from the perspectives of individual users and requests. We finally discuss a broader horizon of ReCCRE: Is it possible to draw a full landscape of evolving human-LLM interaction from observations, modeling the complete evolution trends in the user community as a whole?

5.2.1 Setup

We show that, with metrics as simple as Lexical Richness, we are poised to delineate the climate in great detail and discover the subtleties therein. Specifically, we use MTLD (McCarthy, 2005; McCarthy and Jarvis, 2010) as it's almost fully length-invariant and suitable for random collections. The intuitive interpretation of batched Lexical Richness is a measurement of expression diversity across all users: A higher richness indicates that there are more distinct presentations of requests, while a lower richness indicates converged, collective ways of interactions and clearer system affordances.

Comparing long-term and lay users While our analyses have focused on long-term users for the legitimate comparisons over time, the holistic view here enables us to compare the experience of the "regulars" and the entire public user base. The MTLD data for the full dataset with all 18,964

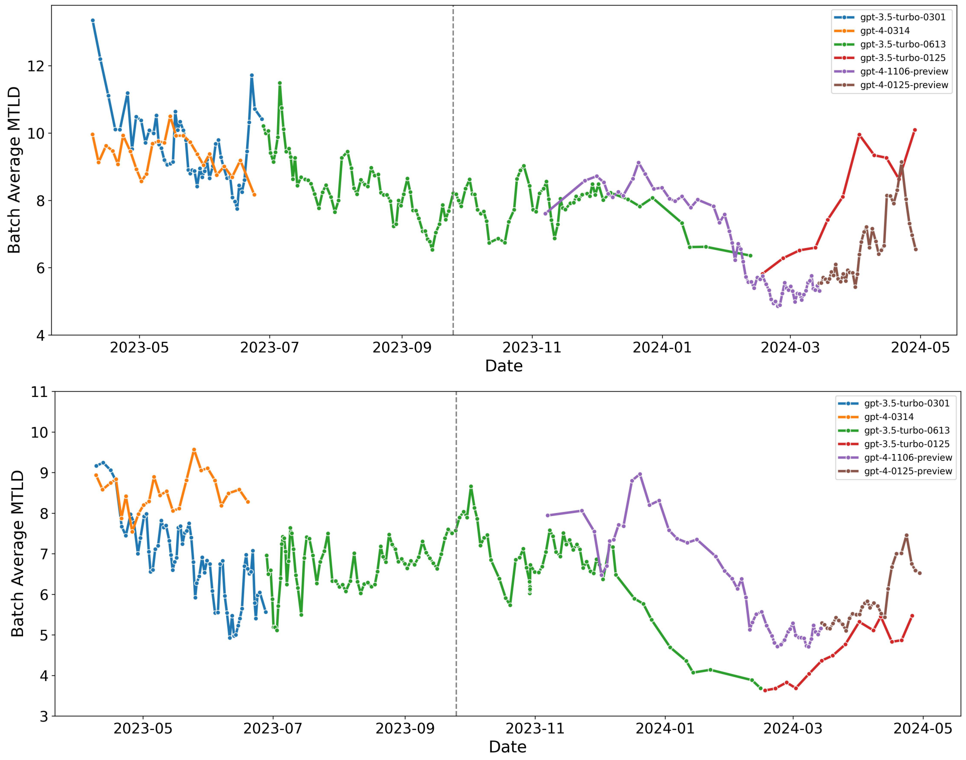


Figure 8: Lexical Richness of long-term users (upper) and all users (lower) across the full time span.

users is illustrated in Figure 8, where proportionally larger batch sizes are used to create comparable densities of data points in the two figures.

5.2.2 Observations

Both long-term and all users share a similar overall trend. The initial deployment of WildChat saw heavily heterogenuous attempts, especially in users who later became long-term. After initial oscillations, the diversity level of expressions stabilized in the extended period of `gpt-3.5-turbo-0613`. However, as the most advanced `gpt-4-1106-preview` was introduced, there seemed a paradigm shift towards a new low. The trend is further different with another major model replacement around March 2024, and both groups see drastically more diverse expressions.

The impact of new models involves a strong pattern of “passing on”: new models, especially the ones from the same family (prefix), take on the exact use patterns of their predecessors. Simultaneous models also see shared evolutions: for instance, the emerging `gpt-4-1106-preview` leads users to also interact with `gpt-3.5-turbo-0613` with less diverse expressions, though the latter is no different from before. More interestingly, long-term users and the lay public seem to show flipped perceptions of model capabilities and usages, as seen in the earliest and latest parts of WildChat. These indicate further complexities yet to be explored, regarding the perception of LLMs in relation to familiarity.

6 Conclusions

While the specificity of user queries is important, *how* users form queries is also significant. We introduce a new task, segmenting requests, context-specific information, assigned roles, and other forms of expression in recorded chat logs. This segmentation allows us to identify patterns both within user experience trajectories and across user populations.

We see strong effects as users gain experience. Our strongest finding is that users tend to move away from making simple requests to combining requests with context-specific information, such as examples or text to be analyzed. Despite well-publicized examples, in all cases users tend to interact with the LLM more as a tool or a search engine and less like a human collaborator.

For model builders, understanding how users learn about and react to the capabilities of models provides clues about how to frame interactions. Our findings also suggest that changes in models actively affect how users behave.

Limitations

While our work seeks to provide new resources and paradigms, the data and analysis work both have practical limitations. Due to the limited capabilities of the 14B LM and the author as annotator and validator, the ReCCRE dataset is selected from the chat logs with English as the major language in the original dataset. This limits the reliable scope of the work, as cultural and language factors can lead to different interaction patterns. Further, our work is based fully on WildChat, which represents a specific type of data collection practice and a specific time window; it is possible that findings are different in interesting ways in other documented chat logs, such as LMSys (Zheng et al., 2024a) collected earlier with a different protocol. In our analyses, we largely utilize off-the-shelf metrics including the *gte* embeddings and Lexical Richness measurements. This is by design, as we would like to demonstrate the usability and low barrier of the dataset. However, we also note that this might limit the boundaries and depth of data analyses, and we encourage further explorations with the ReCCRE data, e.g., fine-tuning models with the resource.

References

Jingwen Cheng, Kshitish Ghate, Wenyue Hua, William Yang Wang, Hong Shen, and Fei Fang. 2025. Realm: A dataset of real-world llm use cases. *arXiv preprint arXiv:2503.18792*.

Alexander S. Choi, Syeda Sabrina Akter, JP Singh, and Antonios Anastasopoulos. 2024. The LLM effect: Are humans truly using LLMs, or are they being influenced by them instead? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22032–22054, Miami, Florida, USA. Association for Computational Linguistics.

Michael A Covington and Joe D McFall. 2010. Cutting the gordian knot: The moving-average type–token ratio (mattr). *Journal of quantitative linguistics*, 17(2):94–100.

Cristian Danescu-Niculescu-Mizil, Moritz Sudhof, Dan Jurafsky, Jure Leskovec, and Christopher Potts. 2013. A computational approach to politeness with application to social factors. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 250–259, Sofia, Bulgaria. Association for Computational Linguistics.

Jie Gao, Simret Araya Gebreegziabher, Kenny Tsu Wei Choo, Toby Jia-Jun Li, Simon Tangi Perrault, and Thomas W Malone. 2024. A taxonomy for human-llm interaction modes: An initial exploration. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, CHI EA '24, New York, NY, USA. Association for Computing Machinery.

Kunal Handa, Alex Tamkin, Miles McCain, Saffron Huang, Esin Durmus, Sarah Heck, Jared Mueller, Jerry Hong, Stuart Ritchie, Tim Belonax, and 1 others. 2025. Which economic tasks are performed with ai? evidence from millions of claude conversations. *arXiv preprint arXiv:2503.04761*.

Zeyu He, Saniya Naphade, and Ting-Hao Kenneth Huang. 2025. Prompting in the dark: Assessing human performance in prompt engineering for data labeling when gold labels are absent. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI '25, New York, NY, USA. Association for Computing Machinery.

Dun-Ming Huang, Pol Van Rijn, Ilia Sucholutsky, Raja Marjieh, and Nori Jacoby. 2024. Characterizing similarities and divergences in conversational tones in humans and LLMs by sampling with people. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10486–10512, Bangkok, Thailand. Association for Computational Linguistics.

Saffron Huang, Esin Durmus, Miles McCain, Kunal Handa, Alex Tamkin, Jerry Hong, Michael Stern, Arushi Somani, Xiuruo Zhang, and Deep Ganguli. 2025. Values in the wild: Discovering and analyzing values in real-world language model interactions. *arXiv preprint arXiv:2504.15236*.

Jonathan Ivey, Shivani Kumar, Jiayu Liu, Hua Shen, Sushrita Rakshit, Rohan Raju, Haotian Zhang, Aparna Ananthasubramaniam, Junghwan Kim, Bowen Yi, and 1 others. 2024. Real or robotic? assessing whether llms accurately simulate qualities of human responses in dialogue. *arXiv preprint arXiv:2409.08330*.

Zhihua Jin, Shiyi Liu, Haotian Li, Xun Zhao, and Huamin Qu. 2025. Jailbreakhunter: a visual analytics approach for jailbreak prompts discovery from large-scale human-llm conversational datasets. *IEEE Transactions on Visualization and Computer Graphics*.

Hannah Rose Kirk, Alexander Whitefield, Paul Rottger, Andrew M Bean, Katerina Margatina, Rafael Mosquera-Gomez, Juan Ciro, Max Bartolo, Adina Williams, He He, and 1 others. 2024. The prism alignment dataset: What participatory, representative and individualised human feedback reveals about the subjective and multicultural alignment of large language models. *Advances in Neural Information Processing Systems*, 37:105236–105344.

Steven Kolawole, Keshav Santhanam, Virginia Smith, and Pratiksha Thaker. 2025. Parallelprompt: Extracting parallelism from large language model queries. *arXiv preprint arXiv:2506.18728*.

Batia Laufer and Paul Nation. 1995. Vocabulary size and use: Lexical richness in l2 written production. *Applied linguistics*, 16(3):307–322.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Zilin Ma, Yiyang Mei, Krzysztof Z. Gajos, and Ian Arawjo. 2024. Schrödinger's update: User perceptions of uncertainties in proprietary large language model updates. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, CHI EA '24, New York, NY, USA. Association for Computing Machinery.

Philip M McCarthy. 2005. *An assessment of the range and usefulness of lexical diversity measures and the potential of the measure of textual, lexical diversity (MTLD)*. Ph.D. thesis, The University of Memphis.

Philip M McCarthy and Scott Jarvis. 2010. Mtld, vocd-d, and hd-d: A validation study of sophisticated approaches to lexical diversity assessment. *Behavior research methods*, 42(2):381–392.

Niloofar Mireshghallah, Maria Antoniak, Yash More, Yejin Choi, and Golnoosh Farnadi. 2024. Trust no bot: Discovering personal disclosures in human-llm conversations in the wild. In *First Conference on Language Modeling*.

Kshitij Mishra, Mauajama Firdaus, and Asif Ekbal. 2022. Please be polite: Towards building a politeness adaptive dialogue system for goal-oriented conversations. *Neurocomputing*, 494:242–254.

Terran Mott, Aaron Fanganello, and Tom Williams. 2024. What a thing to say! which linguistic politeness strategies should robots use in noncompliance interactions? In *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, pages 501–510.

Sheshera Mysore, Debarati Das, Hancheng Cao, and Bahareh Sarrafzadeh. 2025. Prototypical human-ai collaboration behaviors from llm-assisted writing in the wild. *arXiv preprint arXiv:2505.16023*.

Clifford Nass and Youngme Moon. 2000. Machines and mindlessness: Social responses to computers. *Journal of social issues*, 56(1):81–103.

Hope Schroeder, Marianne Aubin Le Quéré, Casey Randazzo, David Mimno, and Sarita Schoenebeck. 2024. Large language models in qualitative research: Can we do the data justice? *arXiv e-prints*, pages arXiv–2410.

Lucas Shen. 2022. LexicalRichness: A small module to compute textual lexical richness.

Alex Tamkin, Miles McCain, Kunal Handa, Esin Durmus, Liane Lovitt, Ankur Rathi, Saffron Huang, Alfred Mountfield, Jerry Hong, Stuart Ritchie, and 1 others. 2024. Clio: Privacy-preserving insights into real-world ai use. *arXiv preprint arXiv:2412.13678*.

Yingfan Wang, Haiyang Huang, Cynthia Rudin, and Yaron Shaposhnik. 2021. Understanding how dimension reduction tools work: An empirical approach to deciphering t-sne, umap, trimap, and pacmap for data visualization. *Journal of Machine Learning Research*, 22(201):1–73.

J.D. Zamfirescu-Pereira, Richmond Y. Wong, Bjoern Hartmann, and Qian Yang. 2023. Why johnny can't prompt: How non-ai experts try (and fail) to design llm prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, New York, NY, USA. Association for Computing Machinery.

Chao Zhang, Shengqi Zhu, Xinyu Yang, Yu-Chia Tseng, Shenrong Jiang, and Jeffrey M Rzeszotarski. 2025a. Navigating the fog: How university students recalibrate sensemaking practices to address plausible falsehoods in llm outputs. In *Proceedings of the 7th ACM Conference on Conversational User Interfaces*, pages 1–15.

Zhouqing Zhang, Kongmeng Liew, and Tham Piumsomboon. 2025b. What one million prompts tells us about ai usage, topics, and preferences. In *2025 IEEE Conference on Artificial Intelligence (CAI)*, pages 174–179. IEEE.

Wenting Zhao, Xiang Ren, Jack Hessel, Claire Cardie, Yejin Choi, and Yuntian Deng. 2024. Wildchat: 1m chatgpt interaction logs in the wild. In *The Twelfth International Conference on Learning Representations*.

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

Mingqian Zheng, Jiaxin Pei, Lajanugen Logeswaran, Moontae Lee, and David Jurgens. 2024b. When "a helpful assistant" is not really helpful: Personas in system prompts do not improve performances of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 15126–15154, Miami, Florida, USA. Association for Computational Linguistics.

Shengqi Zhu, Jeffrey M Rzeszotarski, and David Mimno. 2025. Data paradigms in the era of llms: On the opportunities and challenges of qualitative data in the wild. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, pages 1–8.

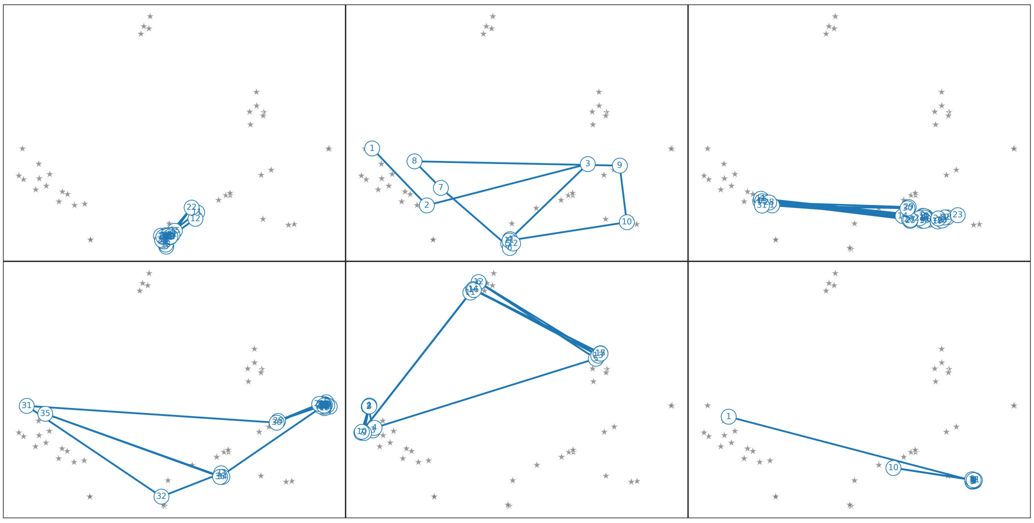


Figure 9: Trajectories of 6 long-term users on the same visualized 2D space as Figure 4.

A Individual Trajectories of Random Users

See Figure 9.

B Prompts for LLM annotators

System You are an expert in linguistics and NLP, and we are working on a project studying how people interact with Large Language Models to chat, make requests, deliver commands, etc. You will annotate some real-world user inputs collected with users' knowledge and content. As you might imagine, people have been talking with these models in drastically varied (and sometimes weird) ways, and for all sorts of purposes. The goal of this project is to create basic annotations of the *components in a user input*.

Prompt First, read through the user input and determine whether this input takes the form of:

- Performing tasks: "Can you write a story about...", "Make this email more polite", "create a list of items", etc.
  - Generating new text or code usually counts towards this type, e.g., summarize", "translate", "brainstorm", etc.
- or, Finding answers: "do I need a visa", "What's the tallest building in the world?", "what happened to Elsa", etc.
  - This usually corresponds to retrieving a clear, short answer, e.g., "give me a few character names that appeared in [book name]".
- or, both
- or, neither

Note that these also apply to briefer inputs that omit the verb, e.g., just "a list of names" as input, instead of "[create] a list of names", similarly belongs to "performing tasks".

Next, your main task is to highlight three types of components that specify the user's demand. Print out the input sentence, with annotations of the following components using "__[...]__", i.e., adding "__[" at the start of a span and "]__" at the end. Additionally, determine which type it is, and add it immediately after "]__", so an annotated piece look like "__[test input snippet]__(ROLE)".

- Assigned Roles: Any roles that the LLM is asked to take on.
  - This includes both the explicit instructions ("You should act like an expert in math", "From now on you will be Alan, my personal assistant") and implicit role assigning ("Does this make sense from a college student's view?", "Because I need the advice from an expert, I'm here to ask about...").
  - Represent assigned roles with "(ROLE)".
- Request: The core span of text that specifies what task(s) exactly the user wants the LLM to perform, or what issue the user wants the LLM to check and answer, no matter how they are phrased.
  - For instance, "write a poem" is the actual request content in both "Hey can you write a poem?" and "Write a poem!",

though the same request is delivered in different tones.
  - Another example: "Can I eat food from 2 days ago that hasn't been fridged?", here the user wants the LLM to look into his plan to "eat food from 2 days ago that hasn't been fridge".
  - Requests are marked as "(REQUEST)".

- Detailed Context: All the detailed context for a given request/question.
  - For example: "Please write a story about two friends. (I can't think of a good one for my kids) ..." – "Write a story about two friends" would be Requests as above, and then the background information "I can't think of a good one for my kids" will be annotated as Detailed Context.
  - The text quoted or pasted for the LLM to process, for instance, the article that goes after the command to "Summarize this article: ", usually counts as detailed context.
  - Detailed Context is annotated as "(CONTEXT)".

Check the text carefully and annotate all the snippets of text that match the three types. Meanwhile, note that this task doesn't require you to give every word a label – some parts of the text are not these three kinds of components, which is an expected result of the task. Generally, the remaining text that isn't annotated with any type should look like a "generic" template that does not involve the information specific to this one instance. If you think the input is a broken instance and cannot be interpreted, or if the input is rather toxic or disturbing, print "__[NONE]__" as the sole output.

Below are some examples:

(3 In-context emamples here)

Now, annotate the following instance:

Input:

(Input text to be annotated)

Output: