Measuring Distribution Shift in User Prompts and Its Effects on LLM Performance

Parker Seegmiller and Sarah Masud Preum
Department of Computer Science
Dartmouth College
Hanover, NH, USA
{pkseeg.gr, Sarah.Masud.Preum}@dartmouth.edu

Abstract

LLMs are increasingly deployed in dynamic, real-world settings, where the distribution of user prompts can shift substantially over time as new tasks, prompts, and users are introduced to a deployed model. Such **natural prompt distribution shift** poses a major challenge to LLM reliability, particularly for specialized models designed for narrow domains or user populations. Despite attention to out-of-distribution robustness, there is very limited exploration of **measuring** natural prompt distribution shift in prior work, and its impact on deployed LLMs remains poorly understood. We introduce the **L**LM **E**valuation under **N**atural prompt **S**hift (**LENS**) framework: a data-centric approach for quantifying natural prompt distribution shift and evaluating its effect on the performance of deployed LLMs. We perform a large-scale evaluation using 192 real-world post-deployment prompt shift settings over time, user group, and geographic axes, training a total of 81 models on 4.68M training prompts, and evaluating on 57.6k prompts. We find that even moderate shifts in user prompt behavior correspond with large performance drops (73% average loss) in deployed LLMs. This performance degradation is particularly prevalent when users from different latent groups and geographic regions interact with models and is correlated with natural prompt distribution shift over time. We systematically characterize how **LLM instruction following ability degrades over time and between user groups**. Our findings highlight the critical need for data-driven monitoring to ensure LLM performance remains stable across diverse and evolving user populations.

1 Introduction

The way humans interact with large language models (LLMs) evolves naturally over time, location, and user groups (Ma et al., 2024; Tafreshipour et al., 2025; Seegmiller et al., 2024). This covariate distribution shift can substantially affect model reliability and safety (Yuan et al., 2023; Myntti et al., 2025; Lazaridou et al., 2021), thereby affecting user satisfaction (Lin et al., 2024b) and overall model performance (Yuan et al., 2023; Yang et al., 2023). We focus on a specific type of covariate shift, which we refer to as **natural prompt distribution shift**: measurable change in the natural distribution of user prompts encountered by a model after deployment, relative to the distribution of user prompts seen during instruction fine-tuning.

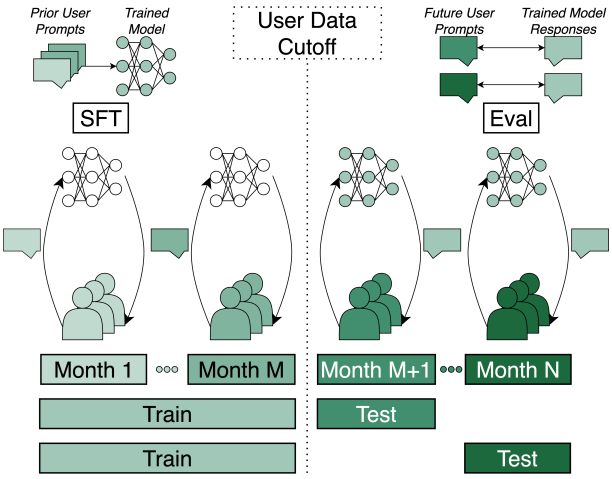

Figure 1: LLM user dynamics change considerably over time, affecting prompt distributions and performance of models trained on prior user prompts. We measure these natural prompt distribution shifts and study their effects on LLM performance.

Real-world user prompts are often repurposed as instruction tuning data, as LLMs fine-tuned on such prompts and collected responses have demonstrated strong performance in downstream deployments (OLMo et al., 2024; Ivison et al., 2023; Cui et al., 2024). However, as with any inputs into a deployed ML system, these prompts are subject to distributional shift from training to deployment (see Figure 1) (Jang et al., 2024; Luu et al., 2022). This is particularly prevalent for smaller LLMs with dedicated user bases in deployment, such as those deployed for customer service (Xu et al., 2024) or

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 32685–32704

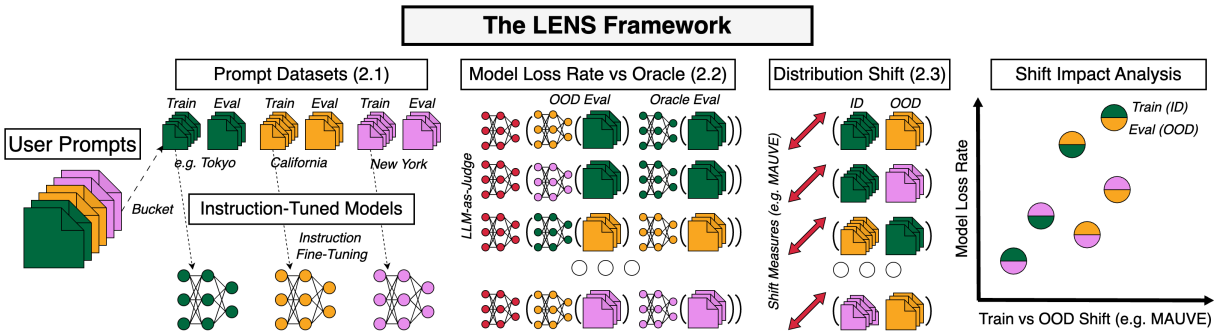

Figure 2: The LENS framework for investigating the relationship between instruction tuning dataset distributions and user prompt distributions in realistic deployment settings. LENS creates {ID/OOD} settings over time, between user groups, and across geographic regions (Section 2.1). LLM OOD performance (Section 2.2) is compared with unsupervised measures of distributional shift (Section 2.3) between prompts seen during instruction fine-tuning vs evaluation. LENS enables analysis of the impact of distributional shift on LLM performance.

medical dialogue (García-Ferrero et al., 2024), as dynamics such as the introduction of new users may affect the distribution of user prompts. If LLMs are not robust to such natural shifts in user prompts after deployment, this can have disastrous effects on user satisfaction and model reliability (Massenon et al., 2025; Wang et al., 2024a).

Despite the practical significance of natural prompt distribution shift and its effects, robust measurement and analysis of the impact of such shifts on deployed LLM performance remain open challenges. Evaluation of LLMs under prompt distribution shift can be viewed as out-of-distribution (OOD) robustness, a well-studied challenge in machine learning (Hendrycks et al., 2021, 2020). There is some evidence that LLMs fail to generalize across domains in specific tasks (Yuan et al., 2023; Wenzel et al., 2022; Teney et al., 2023; Wu et al., 2025b). For example, Wu et al. (2025a) show that natural semantic evolution of context paragraphs from pretraining data may lead to up to 70% decline in LLM reading comprehension performance. Similarly, Yang et al. (2023) show that LLMs frequently experience 20% performance drop in OOD evaluations on GLUE benchmark tasks.

However, evaluating LLM OOD robustness to "in the wild" user prompt distribution shift has yet to be studied at scale. This is a challenge for two reasons. First, distribution shift is not trivial to define or measure for massive-scale corpora of natural language such as user prompts. In prior evaluations of LLM robustness, distribution shift is implicitly assumed rather than measured (Sun et al., 2024; Hupkes et al., 2022; Yuan et al., 2023; Gupta et al., 2024; Teney et al., 2023). Second, the extent to which intentional transfer learning from rounds of pre-training mitigates post-trained LLM performance degradation due to semantic and stylistic divergence between training prompts and those seen in deployment is an open question (Zhao et al., 2024c; Jia, 2024). Robust measurement of natural prompt distribution shifts, and controlled analysis of their impact on LLM performance, will enable a deeper understanding of real-world dynamics affecting LLM evaluation and help to develop more robust model performance *in the wild*.

To address these open challenges, we study two core research questions in this paper. RQ1: How much prompt distribution shift can be observed in the wild? RQ2: How do natural prompt distribution shifts affect performance of pre-trained LLMs?

We introduce the **L**LM **E**valuation under **N**atural prompt distribution **S**hift (**LENS**) framework for analyzing the relationship between instruction tuning dataset distributions and user prompt distributions in deployment. By sourcing data from a large collection of real-world user prompts submitted to a deployed LLM API (Zhao et al., 2024b), the LENS framework pairs 1) real-world user prompts and teacher model responses as instruction tuning data with 2) real-world user prompts in plausible post-deployment shift settings as evaluation data. By using a suite of quantitative measures to estimate various aspects of distribution shift between instruction tuning and evaluation data, and by measuring model performance on shifted evaluation prompts after instruction tuning, this framework offers a systematic, controlled evaluation of model performance across multiple types and degrees of natural shifts in user prompt distributions. Using this evaluation framework, we experiment in 192 post-deployment prompt drift settings, training a

total of 81 models on 4.68M training prompts, and evaluating on 57.6k total evaluation prompts, to evaluate how real-world natural prompt distribution shifts affect deployed LLM performance.

Summary of Findings: Using a suite of unsupervised measures of distributional shift in natural language, we answer RQ1 by finding that natural prompt distributions shift significantly over time, between user groups, and across geographical regions (e.g. average MAUVE of 0.54 between time-delayed prompts and 0.05 between geography-separated prompts). Answering RQ2, we show that LLM performance is considerably worse in naturally-occurring OOD user prompt shift settings[1]. We show that natural prompt distribution shift and LLM performance degradation both increase over time (see Figure 4), and LLM performance degradation is drastic across user groups (88% average loss rate vs oracle models trained on OOD data, see Table 2). Our findings show that instruction following ability learned from real-world user prompts does not naturally transfer in dynamic deployment settings. These insights motivate the need for a better understanding of natural shifts in user-submitted LLM prompts (see Table 3), and more informed methods for mitigating performance loss due to the natural evolution of user prompts.

2 Analysis Methods

In this section we introduce our primary analysis method, the the **L**LM **E**valuation under **N**atural prompt distribution **S**hift (**LENS**) framework. Figure 2 gives an overview of the LENS evaluation framework. First, LENS emulates real-world LLM environments by creating 192 {ID, OOD} (in-distribution, out-of-distribution) dataset splits sourced from real user prompts (Section 2.1). Next, LENS evaluates LLM performance degradation under distributional shift by comparing 81 fine-tuned ID models against fine-tuned OOD (*oracle*) models for each dataset split (Section 2.2), to answer RQ2. Finally, LENS deepens understanding of LLM robustness in real-world settings by estimating the degree of natural prompt distribution shift between ID training and OOD evaluation data in each setting using natural language distribution shift measures (Section 2.3), addressing RQ1. We examine the relationships between natural prompt distribution shift and LLM robustness across a variety of LLMs in Section 3.

[1]We refer to natural prompt shifts as OOD settings.

2.1 Curating Prompt Shift Datasets

We are interested in evaluating LLMs in the context of instruction following post-deployment, so we evaluate LLMs on actual user prompts which they might encounter post-deployment (Song et al., 2019; Don-Yehiya et al., 2025). We do this by sourcing training and evaluation data from LLM queries collected "in the wild." Our goal is to simulate realistic LLM deployment where models go through instruction fine-tuning on previously-collected user prompts and are deployed in evolving environments. As time passes, existing users may shift their behavior, and new users — often from different demographic, geographic, or usage contexts — begin interacting with the model. These dynamics naturally lead to changes in the prompt distribution over time, between user groups, and across regions.

We utilize WildChat (Zhao et al., 2024b), which is comprised of 1M user queries submitted to various OpenAI models, to investigate natural prompt distribution shift in a deployed LLM setting. WildChat contains real-world, natural user prompts in a deployed LLM setting. We use the user prompts and frontier model responses as instruction tuning data and evaluation data (Lin et al., 2024a). We restrict to English prompts for simplicity, and we use the initial user message (prompt) and model response pairs. These prompts were collected by offering users free access to OpenAI models in exchange for consent to use their data. As opposed to other datasets which capture user-generated LLM queries/prompts (Zheng et al.; Köpf et al., 2023; Don-Yehiya et al., 2025), WildChat is uniquely well-suited for LENS as its data collection strategy includes IP addresses and request headers that contain timestamp metadata. Other work has similarly focused on WildChat to explore real-world LLM behavior, including user interaction trends and model response analysis (Brigham et al.; Brahman et al., 2024; Röttger et al., 2025; Zhong et al., 2024; Ye et al., 2024).

We construct 192 {ID, OOD} settings, including 27 unique ID datasets, using prompts and responses from WildChat in which user prompts are bucketed by time, user group, and geographical location (see Table 1). ID data is used for training each model and OOD data is used for evaluating that model in a realistic setting. Oracle training data is drawn from the same bucket as OOD evaluation data (en-

Axis	OOD Eval Data	Oracle Data	ID Data	# ID	# Settings
Time	Month N	Months $<= N$	Months $<= M$, $M < N$	10	64
Example	*Month 5*	*Months 0-5*	*Months 0-2*		
User Group	Users A_i, V_j	Users A_i, V_j	A_k, V_l, $k \neq i$ or $l \neq j$	8	56
Example	*Late/High*	*Late/High*	*Early/Low*		
Geography	Location L_i	Location L_i	Location L_j, $i \neq j$	9	72
Example	*California*	*California*	*Tokyo*		
Total	**19.2k Prompts**	**1.56M Prompts**	**1.56M Prompts**	**27**	**192**

Table 1: Overview of dataset splits created along each axis, resulting in 192 total realistic {ID, OOD} scenarios across 27 unique ID training datasets. As an example {ID, OOD} dataset split by user group, LENS creates ID training data from prompts collected from users who were early adopters with low volume (first 33% of users appearing in WildChat, bottom 33% of users by prompts per day). This is paired with OOD evaluation data collected from users who were late adopters with high volume. Oracle training data in each case contains no overlap with OOD evaluation data. Refer to Appendix A for full dataset creation details.

suring no overlap with OOD data), and is used for training an oracle model whose responses are compared with ID model responses. Details of dataset construction can be found in Appendix A[2].

Over Time. WildChat contains user queries collected from April 2023 to May 2024. As some time periods have more data than others, we sort by timestamp and split the prompts into 12 buckets which we refer to as "months" for simplicity. OOD evaluation datasets consist of 1k prompts sampled exclusively from each month. To simulate training on prior user queries, we construct ID training datasets by randomly sampling prompts from data up to each month. For each OOD evaluation month N, the corresponding oracle data is collected from up to and including month N, indicating training on all available data up to that point. ID data is collected on up to month M where $M < N$, representing different lags in training data availability. This setup mimics a realistic deployment pipeline in which an LLM is trained on past prompts and then deployed to handle future user queries, potentially subject to natural prompt distribution shift.

Between User Groups. We treat anonymized IP addresses as proxies for unique users, which is a realistic setting for deployed models which often operate without persistent user accounts (Ye et al., 2024). As most individual users do not submit a sufficient number of prompts to train a model or analyze performance trends, we divide users into user groups based on two latent behavioral dimensions: Adoption stage (early, medium, late), determined by when a user first appeared in the dataset; and query volume (low, medium, high), based on the rate of queries submitted per day. This results in 8 usable user groups (excluding late adopters with low volume, due to insufficient data). For each group, we sample 10k training and 1k evaluation prompts. We consider evaluation scenarios in which oracle training and OOD evaluation data come from the same user group, and ID training data comes from a different group, isolating the effect of new user groups on model performance.

Across Locations. WildChat uses IP-to-location mappings to assign each query a geographic region. We consider city/state-level granularity and select 9 regions with sufficient data coverage (5k for training, 1k for evaluation). These states serve as coarse representations of distinct geographical regions, e.g. California, Paris and Tokyo. We consider evaluation scenarios in which oracle training and OOD evaluation data come from the same location, and ID training data comes from a different location, allowing us to assess performance under natural geographic prompt distribution shift.

2.2 Evaluating LLM OOD Performance

LLM performance can be influenced by a wide variety of factors including pretraining data (Myntti et al., 2025), dataset size (Kaplan et al., 2020), and training techniques (Zhang et al.). In studying OOD performance, we wish to isolate the effects of natural prompt distribution shift between instruction fine-tuning data and evaluation data. We do this by fixing a base model (and thus pretraining data/procedures), instruction tuning dataset size, training procedures, and model evaluation procedures. We give an overview of these processes here, and give full details in Appendix B.

We use three pre-trained base models in our ex-

[2]Dataset creation code can be found at https://github.com/pkseeg/lens.

periments. We first use Qwen2.5-7B[3] (Qwen et al., 2025), as this model is open-source and has been used for other work investigating instruction fine-tuning (Chen et al., 2024; Dong et al., 2025). To investigate the relationship between model size and model degradation under natural prompt distribution shift, we employ the larger Qwen2.5-14B[4]. Finally, to compare the effects of natural prompt distribution shift on a model from a different model family, we use Llama3-8B[5] (AI@Meta, 2024), which has similarly been used to investigate instruction fine-tuning (Zheng et al., 2024; Li et al., 2024). As we train 3 base models in each of 27 ID settings (see Table 1), we train 81 models total. All three models are considerably smaller than the models used to generate responses in WildChat, meaning WildChat is viable as both a learning objective, i.e. distillation of the OpenAI models in WildChat, and an evaluation setting (Lin et al., 2024a). As we focus on measuring natural shifts in user prompts, we choose to scale {ID, OOD} dataset pairs (rather than models) to get a more comprehensive view of possible natural prompt shift settings, and use three models from different sizes and families to evaluate performance under these shifts. Thus our evaluation prioritizes computational resources to answer research questions about the types and impact of shift in broad settings, rather than detailed comparisons between many different LLMs.

In each setting, we perform supervised fine-tuning of each base model to create two fine-tuned models. For each dataset series using each base model, we train an ID model on the ID dataset and an oracle model on the oracle dataset[6]. We directly compare the ID and oracle models by generating responses for 100 fixed prompts from the OOD evaluation data in each of the 192 splits (19.2k total for each base model). Using the same evaluation prompt as in WildChat (Zhao et al., 2024b), we compare ID and oracle model responses using GPT-4o as the judge LLM (Zheng et al., 2023). We evaluate the performance of the ID model on OOD evaluation data by using the ID model loss rate (conversely oracle model win rate), i.e. percentage of response comparisons won by the model trained on oracle instruction tuning pairs. Thus, ID model loss rate indicates the extent to which sourcing training data from the OOD setting is better than ID (i.e. excluding ties). For example, a Qwen2.5-7B model trained on data from months 0-3 loses 53% of head-to-head comparisons on OOD evaluation data in month 7 against an oracle model trained on data from months 0-7, and ties 15%, hence the model is winning only 30% of these comparisons and is worse than the oracle model. The greater the ID model loss rate, the greater the disparity between the ID and oracle models, thus the less performant the ID data in deployment. We follow other works in using LLM performance as the evaluation metric for training data quality (Sun et al., 2024; Zhuang et al., 2025; Chen et al., 2023).

[3]Qwen/Qwen2.5-7B
[4]Qwen/Qwen2.5-14B
[5]meta-llama/Meta-Llama-3-8B
[6]Training details can be found in Appendix B.

2.3 Measures of Natural Prompt Distribution Shift

Among large-scale robustness evaluations of LLMs (Yuan et al., 2023; Wenzel et al., 2022), a critical missing element is the degree to which the OOD setting differs from the training setting. Not all distribution shifts are created equal, and faulty assumptions of distributional shift may lead to misleading estimates of model robustness. However, measuring distributional shift in natural language presents unique challenges due to the lack of a known ground-truth distribution over text inputs (LeBrun et al., 2022). Unlike structured data with explicit feature spaces, natural language is inherently high-dimensional, sparse, and context-dependent. As a result, quantifying natural prompt distribution shift requires proxy measures that capture meaningful variation between text corpora. Prior work has leveraged supervised approaches to detect shift in known prompt properties such as complexity (He et al., 2024), toxicity (Zheng et al.), or topic (Myntti et al., 2025). However, these methods can be impractical for deployed LLMs, where future prompts may diverge in unforeseen and unconstrained ways (Jang et al., 2024). To address this challenge, the LENS framework emphasizes unsupervised measures of distributional shift in natural language. We briefly describe four measures used in our analysis here, describing full details and additional measures in Appendix C.

Notation Let P denote the distribution of pre-deployment training user prompts, and Q denote the distribution of post-deployment user prompts. We call $X = \{X_i\}_{i=1}^{n} \sim P$ the sample of n training prompts, and $Y = \{Y_i\}_{i=1}^{m} \sim Q$ the sample of m post-deployment prompts. To measure shift between training prompts and post-deployment prompts, we will use sample-based divergence

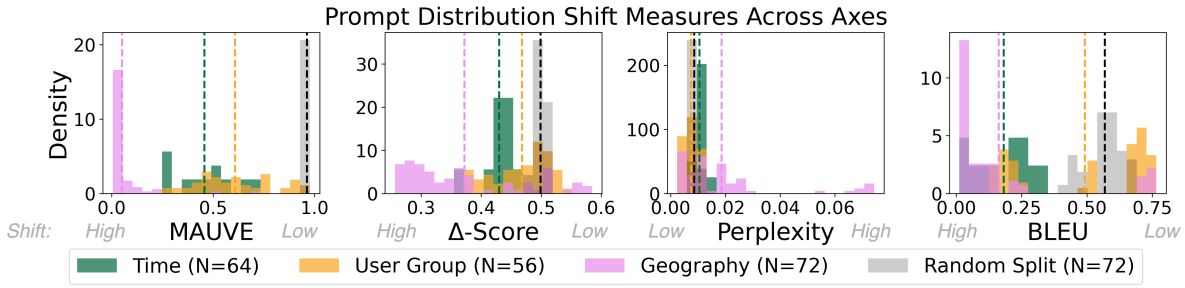

Figure 3: We estimate natural prompt distribution shift between all {ID, OOD} settings using four distribution shift measures, finding that prompt distributions shift significantly over time, between user groups, and across geographical locations. We also find that MAUVE can best distinguish between distributions, as it has a high disparity between estimates across axes and the random baseline estimates, which are more precise and stable.

estimators $D(P, Q)$. In practice, we use $n = m = 1000$ samples from pre- and post-deployment prompt pools (Seegmiller and Preum, 2023). The $n = 1000$ pre-deployment training prompts are taken as a subset from the ID training data and the $m = 1000$ post-deployment prompts are the OOD data described in Section 2.1.

MAUVE, introduced by Pillutla et al. (2021), is an embeddings-based measure designed for text generation evaluation , which compares a pair of text distributions by computing information divergences in a quantized embedding space. MAUVE simultaneously captures the probability of Type I and Type II errors in language generation — thus repurposing MAUVE to measure divergence between prompt distributions $D(P, Q)$ is meant to estimate the overlap in relative frequency of prompt characteristics in P and Q. A MAUVE score closer to 0 indicates higher shift between prompt distributions.

Δ-score, introduced by Seegmiller and Preum (2023),[7] is an embeddings-based measure which uses a similarity-based statistical depth $D_\delta(x_i, P)$ to estimate how representative a text $x_i \sim P$ is of the distribution. The goal of the Δ-score is to capture how well one text distribution represents another. A Δ-score of 0.5 indicates an equal likelihood of representation, and a lower Δ-score indicates higher shift.

Perplexity is a language model-based measure that quantifies how well a probability distribution predicts a sample. Lower perplexity values indicate that the model assigns higher probabilities to the observed sequences, suggesting better predictive performance. The LENS framework utilizes the LLM itself, fine-tuned on samples $\{X_i\}_{i=1}^n \sim P$ to estimate perplexity of samples $\{Y_i\}_{i=1}^n \sim Q$. This approach captures how *surprising* prompts from the Q distribution are based on P.

BLEU, originally developed for machine translation evaluation (Papineni et al., 2002), provides a statistical measure of n-gram overlap between text distributions. Higher BLEU scores indicate greater lexical similarity between distributions, capturing surface-level textual patterns rather than deep semantic relationships. BLEU offers an estimate of lexical divergence between prompt distributions, measuring explicit differences in vocabulary usage, phrase construction, and local contextual patterns.

3 Evaluation and Results

In this section we quantify the magnitude of natural prompt distribution shift in Section 3.1 (RQ1), then we examine the degrading impact of this shift on LLM performance in Sections 3.2 and 3.3 (RQ2). Results in Sections 3.1 and 3.2 are averaged across three models (Llama3-8B, Qwen2.5-7B, and Qwen2.5-14B) to ensure robustness; Section 3.3 provides a detailed inter-model comparison.

3.1 Distribution Shift in Real-World LLM Queries

The LENS framework measures natural prompt distribution shift using the unsupervised measures described in Section 2.3: MAUVE, Δ-score, perplexity, and BLEU. Shifts are estimated between training and OOD evaluation data in each of the 192 {ID, OOD} natural settings described in Section 2.1 along three axes: time, user group, and geographical location. To compare these estimates against a baseline amount of shift which would be expected in random sampling, we also measure shift between 72 dataset pairs randomly sampled from all English WildChat prompts.

[7] The original authors use Q-score as notation, but we use Δ-score for notational clarity in this work.

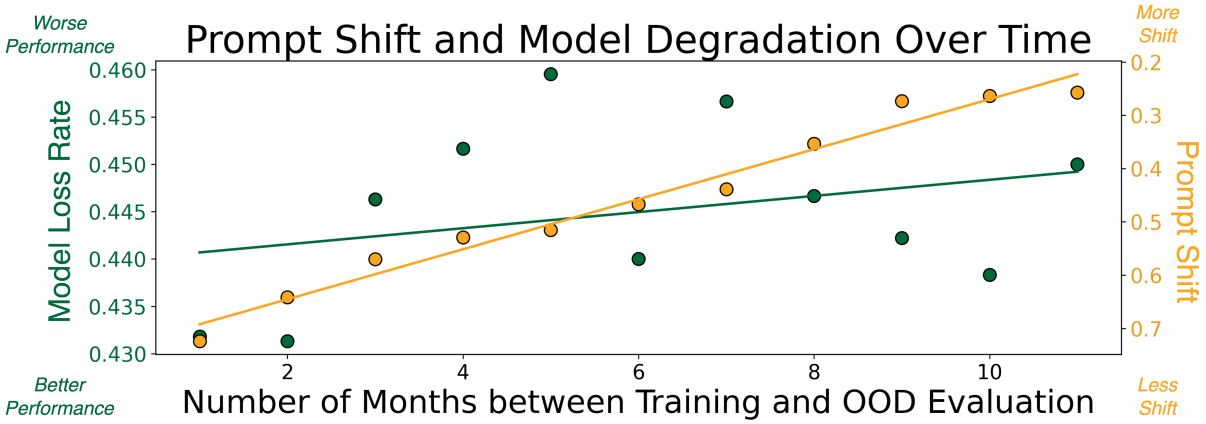

Figure 4: Using the LENS framework to evaluate LLM performance in **64** time-shifted deployment settings, averaging across 3 different LLMs, we find that significant distributional shift (measured in MAUVE) occurs in user prompts submitted to a deployed LLM over time. We find that the degree of **natural prompt distribution shift** increases as the lag between training data and time-delayed OOD evaluation data increases. We also find that LLMs trained on user prompts perform poorly on time-delayed OOD evaluation data (measured in model loss rate vs an oracle model trained on OOD data), and that the performance loss tends to be more drastic as the lag increases. See Section 3.2 for more discussion of these results.

As shown in Figure 3, significant natural prompt distribution shifts are observed across all three axes. Notably, the geographical location axis exhibits the strongest shift, indicating that different geographical behavioral patterns are strongly associated with changes in prompt characteristics. Natural prompt distribution shifts are also strong between time-shifted datasets, e.g. with an average MAUVE score of 0.54 vs the random baseline MAUVE score of 0.96. While user groups based on adoption rate and usage volume exhibit the least natural prompt distribution shift among the three axes which we evaluate, the average MAUVE (0.61), Δ-score (0.47), and BLEU (0.49) scores in this group still indicate measurable distributional shift compared to baseline scores (0.96, 0.50, 0.57 average MAVUE, Δ-score, and BLEU, respectively). These findings highlight that prompt distributions encountered by deployed LLMs can vary substantially depending on when the model is used and who is using it. We find that perplexity estimates shift between datasets to be closer to the random baseline. We hypothesize that perplexity measurements are impacted by the vast amounts of pre-training data seen by the LLM, impacting the surprisal of the model on OOD data. However, as we see in Table 2, pre-training does not mitigate performance degradation in the context of natural prompt distribution shift. Shift estimates can be found in tabular format in Table 4 in Appendix E.

3.2 LLM Performance Degradation Due to Natural Prompt Distribution Shift

The LENS framework evaluates the relationship between natural prompt distribution shift and LLM performance by measuring LLM OOD performance in realistic post-deployment settings. For each prompt in the OOD evaluation dataset, we collect responses from two models: the ID model (i.e., the model trained on data representative of expected pre-deployment user prompts), and the oracle model trained on data sourced from the same bucket as the OOD evaluation dataset. Following Zhao et al. (2024b), we use GPT-4o to judge the responses. We measure ID model loss rate compared to the oracle model to estimate LLM OOD performance degradation.

Figure 4 presents results along the time axis, averaged by the number of months between ID training and OOD evaluation settings. We see a correlation between the magnitude of natural prompt distribution shift and the degradation in LLM performance, both of which increase over time. We see a similar pattern in user group and geography settings, which we detail in Appendix G. Aggregated performance metrics can be found in tabular format in Table 6 in Appendix F.

3.3 Inter-Model Performance Degradation

Table 2 gives distribution shift (MAUVE and Δ-score[8]) and performance (loss, win, and tie rates

[8]Perplexity and BLEU scores are given in Appendix E.

Axis	Model	MAUVE	Δ-score	Loss Rate vs Oracle	Win Rate vs Oracle	Tie Rate vs Oracle
Time	L8B	0.5125 (0.18)	0.4963 (0.02)	0.4388 (0.05)	0.3991 (0.05)	0.1359 (0.03)
Time	Q7B	0.5437 (0.18)	0.4175 (0.04)	0.4409 (0.05)	0.3553 (0.04)	0.1770 (0.04)
Time	Q14B	0.5510 (0.17)	0.4049 (0.05)	0.4509 (0.05)	0.3573 (0.05)	0.1630 (0.04)
UG	L8B	0.5906 (0.20)	0.4666 (0.03)	0.8466 (0.04)	0.0196 (0.02)	0.1166 (0.03)
UG	Q7B	0.6139 (0.19)	0.4689 (0.07)	0.8780 (0.02)	0.0173 (0.01)	0.0880 (0.03)
UG	Q14B	0.6217 (0.18)	0.4685 (0.08)	0.8923 (0.03)	0.0170 (0.01)	0.0730 (0.02)
Geo	L8B	0.0471 (0.10)	0.3642 (0.09)	0.8778 (0.05)	0.0099 (0.01)	0.0917 (0.04)
Geo	Q7B	0.0529 (0.11)	0.3787 (0.11)	0.8849 (0.04)	0.0104 (0.01)	0.0864 (0.04)
Geo	Q14B	0.0546 (0.11)	0.3737 (0.10)	0.8833 (0.04)	0.0107 (0.01)	0.0871 (0.05)

Table 2: MAUVE, Δ-score, and performance against oracle model across models and prompt shift axes. MAUVE (0-1) measures distributional similarity between ID and OOD prompt distributions by estimating Type I and Type II errors in language generation (higher is more similar). Δ-score (0-0.5) measures the likelihood that a random OOD prompt is representative of the ID distribution (0.5 implies no shift, lower is more shift). Loss/Win/Tie Rate reflect the proportion of head-to-head response comparisons in which the model trained on shifted-distribution prompts performs worse than, better than, or equivalently to the oracle model (trained on prompts from the same distribution as the evaluation data). Models: Llama3-8B (L8B), Qwen2.5-7B (Q7B), Qwen2.5-14B (Q14B). Axes: Time, User Group (UG), and Geography (Geo). All three natural shift axes degrade performance, with an average loss rate of 73%, and this degradation is consistent across model architecture and size. Scores are mean (standard deviation).

vs oracle model as described in Section 2.2) results for each model, averaged for each natural prompt shift axis. We see that the prompt shifts found in Section 3.1 are consistent across model families and sizes. With average loss rates of 0.44, 0.88 and 0.88 compared to win rates of 0.37, 0.02, and 0.01 for time, user group, and geographical shift settings, respectively, performance degradation in OOD settings is consistent in all three models, regardless of model size or family. We find that performance drop across user group and geographical splits is particularly high, with average loss rates of 87% and 88%, indicating that instruction following ability generalizes poorly across user groups and geographical splits. These results indicate that instruction following ability learned from real-world user prompts does not transfer in natural prompt shift settings.

3.4 Towards Identifying Underlying Prompt Shift Trends

We have shown that instruction following ability learned from real-world user prompts does not naturally transfer in dynamic deployment settings. To motivate future investigation into the underlying drivers of this natural prompt shift, we conduct an initial qualitative analysis of 30 representative[9] prompt pairs sampled from ID and OOD splits along each axis (90 total). Representative examples from each axis are shown in Table 3.

Over Time. Representative prompts from later time periods tended to be longer and more structurally elaborate, with explicit formatting constraints and instructions, while earlier prompts were more exploratory and casual. These observations are consistent with prior findings that user behavior evolves as familiarity with model capabilities grows (Ma et al., 2024; Tafreshipour et al., 2025), suggesting that temporal prompt shift may reflect changing user expectations rather than purely topical drift.

Between User Groups. Representative prompts across user groups revealed systematic differences in how distinct populations conceptualize and engage with the model. High-volume users, for example, showed evidence of templated, repetitive prompting, suggesting workflow integration rather than exploratory use—for example, variations of the templated ID prompt in row 2 of Table 3 was sampled several times in our manual analysis. Low-volume users exhibited more eclectic, less patterned prompt distributions spanning a wider range of domains and task types. These patterns suggest that user group-level shift may reflect systematic differences in how distinct user populations conceptualize and engage with deployed LLMs, consistent with the high ID model loss rates (avg. 88%) observed for this axis in Section 3.3.

[9]Details of the sample selection and labeling process is given in Appendix D, along with three examples per axis.

Axis	ID Prompt	OOD Prompt
Time	(Month 4) Make up sentences with this word: impartial	(Month 9) As a prompt generator... you will create image prompts for the AI to visualize... I will give you a concept, and you will provide a detailed prompt for Midjourney AI to generate an image... Please adhere to the structure and formatting below, and follow these guidelines...
User Group	(High Volume) give me a response to {message} to send in a discussion, VERY SHORT, CONCISE & CLEAR. ONLY RETURN THE RAW MESSAGE, DO NOT SAY "Hey here is the message you asked"	(Medium Volume) write some Eating disorders that will be taught for nutritional biochemistry course
Geography	(Massachusetts) write a comedic and vividly detailed story set in the TV show Z Nation about 10K ...	(Paris) explique de manière claire pour ma documentation ma pipeline ...

Table 3: Representative ID/OOD prompt pairs illustrating qualitative shift trends along each natural prompt distribution shift axis. We find that prompts instructions tend to become more specific over time, that high volume usage is consistent with more templated prompts, and that geographically distinct prompts contain cultural and linguistic differences.

Across Geographic Regions. In our manual analysis, representative prompts from different geographic regions tended to reflect various cultural differences (see example in Table 3), in addition to linguistic variations potentially stemming from the use of English as a second language or code-mixing (Yang and Chai, 2025).

Our findings motivate future work on systematic taxonomies of prompt shift categories and supervised probes for interpretable shift decomposition. More broadly, our analysis suggests that evaluating and mitigating natural prompt distribution shift is a general challenge for any deployed LLM operating over an evolving user base. Robust evaluation under natural prompt shift will become increasingly important for LLM reliability as deployment continues to scale and users continue to evolve.

4 Related Works

OOD Robustness in LLMs Recent work has argued that many OOD tests do not represent a sufficient probing of LLM robustness (Gupta et al., 2024; Teney et al., 2023). This is a pronounced issue in NLP, where the definition of OOD data can be less clear-cut compared to other machine learning settings, partially because of the difficulty of defining a natural distributional model of natural language (LeBrun et al., 2022; Ilia and Aziz, 2024). Some prior works have presented thorough evaluations of OOD robustness in LLMs by systematically selecting ID and OOD dataset pairs across a wide variety of common tasks, including some in NLP (Yuan et al., 2023; Wenzel et al., 2022). These focus on task-specific OOD robustness, where the LENS framework focuses on OOD robustness to user prompting in deployment and measures distribution shift in an unsupervised manner.

Data-Centric Distribution Shift Evaluation in NLP Prior works have investigated temporal shift in language processing tasks such as document classification, named entity recognition, language modeling, and sentiment classification (Agarwal and Nenkova, 2022; Biesialska et al., 2020; Loureiro et al., 2022; Lazaridou et al., 2021; Huang and Paul, 2018; Jang et al., 2024; Luu et al., 2022). Other works have analyzed LLM performance by assessing the relationships between pre-training data term frequencies and downstream performance (McCoy et al., 2023; Razeghi et al., 2022; Kandpal et al., 2023; Elazar et al., 2022). Other works have investigated robustness of LLMs to changes in instructions (Sun et al., 2024; Mizrahi et al., 2024; Gu et al., 2023). We are the first to measure natural prompt shifts and examine the impacts on LLMs in deployment.

5 Conclusion

We introduced the **L**LM **E**valuation under **N**atural prompt **S**hift (**LENS**) framework for analyzing the relationship between instruction tuning dataset distributions and user prompt distributions in deployment. We find that natural prompt distribution shifts correlate with LLM post-deployment performance degradation. We provide the LENS framework and our findings to facilitate future research in evaluating and mitigating LLM performance loss under natural prompt distribution shift.

6 Limitations

Large scale evaluation of LLMs is expensive: our evaluation required an estimated 1,200 GPU hours on a single A100 between model training, response

generation, and model-based shift measurement. We choose to scale shift settings and models, but this limits our investigation into explicit mixing of distributions for diversity (Yuan et al., 2023), and intersectionality (e.g. specific user groups over time), which may be of interest in specific domains (Devinney et al., 2024). We are limited by the availability of real-world user prompt datasets with temporal and user designations (Zhao et al., 2024b; Zheng et al.; Köpf et al., 2023). We follow WildChat (Zhao et al., 2024b) in using LLM-As-Judge in our evaluation, which is known to have some limitations (Zheng et al., 2023; Krumdick et al., 2025).

Acknowledgments

This work was supported in part by NIH grant 1R21DA059665-01A1, Clinical and Translational Science Institute grant 1UM1TR004772, a Google Research Scholar award, and the NSF Research Traineeship, Transformative Research, and Graduate Education in Sensor Science, Technology and Innovation grant (DGE-2125733).

References

Ahmed Abdulaal, Chen Jin, Nina Montaña-Brown, Aryo Pradipta Gema, Daniel C Castro, Daniel C Alexander, Philip Alexander Teare, Tom Diethe, Dino Oglic, and Amrutha Saseendran. Balancing act: Diversity and consistency in large language model ensembles. In *The Thirteenth International Conference on Learning Representations*.

Oshin Agarwal and Ani Nenkova. 2022. Temporal effects on pre-trained models for language processing tasks. *Transactions of the Association for Computational Linguistics*, 10:904–921.

AI@Meta. 2024. Llama 3 model card.

Magdalena Biesialska, Katarzyna Biesialska, and Marta R Costa-jussà. 2020. Continual lifelong learning in natural language processing: A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6523–6541.

A Binu Jose and Pranesh Das. 2022. A multi-objective approach for inter-cluster and intra-cluster distance analysis for numeric data. In *Soft Computing: Theories and Applications: Proceedings of SoCTA 2021*, pages 319–332. Springer.

Faeze Brahman, Sachin Kumar, Vidhisha Balachandran, Pradeep Dasigi, Valentina Pyatkin, Abhilasha Ravichander, Sarah Wiegreffe, Nouha Dziri, Khyathi Chandu, Jack Hessel, and 1 others. 2024. The art of saying no: Contextual noncompliance in language models. *Advances in Neural Information Processing Systems*, 37:49706–49748.

Natalie Grace Brigham, Chongjiu Gao, Tadayoshi Kohno, Franziska Roesner, and Niloofar Mireshghallah. Developing story: Case studies of generative ai's use in journalism. In *Workshop on Socially Responsible Language Modelling Research*.

Yuheng Bu, Shaofeng Zou, Yingbin Liang, and Venugopal V Veeravalli. 2016. Estimation of kl divergence between large-alphabet distributions. In *2016 IEEE International Symposium on Information Theory (ISIT)*, pages 1118–1122. IEEE.

Hao Chen, Yiming Zhang, Qi Zhang, Hantao Yang, Xiaomeng Hu, Xuetao Ma, Yifan Yanggong, and Junbo Zhao. 2023. Maybe only 0.5% data is needed: A preliminary exploration of low training data instruction tuning. *arXiv preprint arXiv:2305.09246*.

Xinyi Chen, Baohao Liao, Jirui Qi, Panagiotis Eustratiadis, Christof Monz, Arianna Bisazza, and Maarten Rijke. 2024. The sifo benchmark: Investigating the sequential instruction following ability of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1691–1706.

Erfei Cui, Yinan He, Zheng Ma, Zhe Chen, Hao Tian, Weiyun Wang, Kunchang Li, Yi Wang, Wenhai Wang, Xizhou Zhu, Lewei Lu, Tong Lu, Yali Wang, Limin Wang, Yu Qiao, and Jifeng Dai. 2024. Sharegpt-4o: Comprehensive multimodal annotations with gpt-4o.

Hannah Devinney, Jenny Björklund, and Henrik Björklund. 2024. We don't talk about that: Case studies on intersectional analysis of social bias in large language models. In *Proceedings of the 5th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, pages 33–44.

Shachar Don-Yehiya, Leshem Choshen, and Omri Abend. 2025. The sharelm collection and plugin: contributing human-model chats for the benefit of the community. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 167–177.

Guanting Dong, Xiaoshuai Song, Yutao Zhu, Runqi Qiao, Zhicheng Dou, and Ji-Rong Wen. 2025. Toward verifiable instruction-following alignment for retrieval augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23796–23804.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Amir Feder, Abhilasha Ravichander, Marius Mosbach, Yonatan Belinkov, Hinrich Schütze, and Yoav Goldberg. 2022. Measuring causal effects of data statistics on language model'sfactual'predictions. *arXiv preprint arXiv:2207.14251*.

Iker García-Ferrero, Rodrigo Agerri, Aitziber Atutxa, Elena Cabrio, Iker de la Iglesia, Alberto Lavelli, Bernardo Magnini, Benjamin Molinet, Johana

Ramirez-Romero, German Rigau, and 1 others. 2024. Medical mt5: An open-source multilingual text-to-text llm for the medical domain. In *LREC-COLING 2024-2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*.

Jiasheng Gu, Hongyu Zhao, Hanzi Xu, Liangyu Nie, Hongyuan Mei, and Wenpeng Yin. 2023. Robustness of learning from task instructions. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13935–13948.

Ashim Gupta, Rishanth Rajendhran, Nathan Stringham, Vivek Srikumar, and Ana Marasović. 2024. Whispers of doubt amidst echoes of triumph in nlp robustness. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5533–5590.

Xuan He, Da Yin, and Nanyun Peng. 2024. Guiding through complexity: What makes good supervision for hard reasoning tasks? *arXiv preprint arXiv:2410.20533*.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, and 1 others. 2021. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8340–8349.

Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Song. 2020. Pretrained transformers improve out-of-distribution robustness. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2744–2751.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Xiaolei Huang and Michael Paul. 2018. Examining temporality in document classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 694–699.

Dieuwke Hupkes, Mario Giulianelli, Verna Dankers, Mikel Artetxe, Yanai Elazar, Tiago Pimentel, Christos Christodoulopoulos, Karim Lasri, Naomi Saphra, Arabella J. Sinclair, Dennis Ulmer, Florian Schottmann, Khuyagbaatar Batsuren, Kaiser Sun, Koustuv Sinha, Leila Khalatbari, Maria Ryskina, Rita Frieske, Ryan Cotterell, and Zhijing Jin. 2022. A taxonomy and review of generalization research in nlp. *Nature Machine Intelligence*, 5:1161 – 1174.

Evgenia Ilia and Wilker Aziz. 2024. Predict the next word. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 234–255.

Hamish Ivison, Yizhong Wang, Valentina Pyatkin, Nathan Lambert, Matthew E Peters, Pradeep Dasigi, Joel Jang, David Wadden, Noah A Smith, Iz Beltagy, and 1 others. 2023. Camels in a changing climate: Enhancing lm adaptation with tulu 2. *CoRR*.

Myeongjun Jang, Antonios Georgiadis, Yiyun Zhao, and Fran Silavong. 2024. Driftwatch: A tool that automatically detects data drift and extracts representative examples affected by drift. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track)*, pages 335–346.

Robin Jia. 2024. Does distribution shift matter in the era of pre-trained language models? Domain Adaptation and Related Areas Workshop, Simons Institute. Workshop presentation.

Thinking Machines John Schulman. 2025. Lora without regret.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2023. Large language models struggle to learn long-tail knowledge. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 15696–15707. PMLR.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

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.

Michael Krumdick, Charles Lovering, Varshini Reddy, Seth Ebner, and Chris Tanner. 2025. No free labels: Limitations of llm-as-a-judge without human grounding. *arXiv preprint arXiv:2503.05061*.

Angeliki Lazaridou, Adhi Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Tomas Kocisky, Sebastian Ruder, and 1 others. 2021. Mind the gap: Assessing temporal generalization in neural language models. *Advances in Neural Information Processing Systems*, 34:29348–29363.

Benjamin LeBrun, Alessandro Sordoni, and Timothy J O'Donnell. 2022. Evaluating distributional distortion in neural language modeling. In *International Conference on Learning Representations*.

Yoonsang Lee, Pranav Atreya, Xi Ye, and Eunsol Choi. 2024. Crafting in-context examples according to lms' parametric knowledge. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2069–2085.

Ziniu Li, Congliang Chen, Tian Xu, Zeyu Qin, Jiancong Xiao, Zhi-Quan Luo, and Ruoyu Sun. 2024. Preserving diversity in supervised fine-tuning of large language models. In *The Thirteenth International Conference on Learning Representations*.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. 2024a. Wildbench: Benchmarking llms with challenging tasks from real users in the wild. In *The Thirteenth International Conference on Learning Representations*.

Ying-Chun Lin, Jennifer Neville, Jack W Stokes, Longqi Yang, Tara Safavi, Mengting Wan, Scott Counts, Siddharth Suri, Reid Andersen, Xiaofeng Xu, and 1 others. 2024b. Interpretable user satisfaction estimation for conversational systems with large language models. *arXiv preprint arXiv:2403.12388*.

Xiaoxuan Liu, Lanxiang Hu, Peter Bailis, Alvin Cheung, Zhijie Deng, Ion Stoica, and Hao Zhang. 2024. Online speculative decoding. In *Proceedings of the 41st International Conference on Machine Learning*, pages 31131–31146.

Daniel Loureiro, Francesco Barbieri, Leonardo Neves, Luis Espinosa Anke, and Jose Camacho-Collados. 2022. Timelms: Diachronic language models from twitter. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 251–260.

Jinghui Lu, Maeve Henchion, and Brian Mac Namee. 2020. Diverging divergences: Examining variants of jensen shannon divergence for corpus comparison tasks. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6740–6744.

Kelvin Luu, Daniel Khashabi, Suchin Gururangan, Karishma Mandyam, and Noah A Smith. 2022. Time waits for no one! analysis and challenges of temporal misalignment. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5944–5958.

Yihan Ma, Xinyue Shen, Yixin Wu, Boyang Zhang, Michael Backes, and Yang Zhang. 2024. The death and life of great prompts: analyzing the evolution of llm prompts from the structural perspective. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21990–22001.

Rhodes Massenon, Ishaya Gambo, Javed Ali Khan, Christopher Agbonkhese, and Ayed Alwadain. 2025. " my ai is lying to me": User-reported llm hallucinations in ai mobile apps reviews. *Scientific Reports*, 15(1):30397.

R Thomas McCoy, Shunyu Yao, Dan Friedman, Matthew Hardy, and Thomas L Griffiths. 2023. Embers of autoregression: Understanding large language models through the problem they are trained to solve. *arXiv preprint arXiv:2309.13638*.

Moran Mizrahi, Guy Kaplan, Dan Malkin, Rotem Dror, Dafna Shahaf, and Gabriel Stanovsky. 2024. State of what art? a call for multi-prompt llm evaluation. *Transactions of the Association for Computational Linguistics*, 12:933–949.

Amanda Myntti, Erik Henriksson, Veronika Laippala, and Sampo Pyysalo. 2025. Register always matters: Analysis of llm pretraining data through the lens of language variation. *arXiv preprint arXiv:2504.01542*.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, and 1 others. 2024. 2 olmo 2 furious. *arXiv preprint arXiv:2501.00656*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. Mauve: Measuring the gap between neural text and human text using divergence frontiers. *Advances in Neural Information Processing Systems*, 34:4816–4828.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Yasaman Razeghi, Robert L Logan IV, Matt Gardner, and Sameer Singh. 2022. Impact of pretraining term frequencies on few-shot numerical reasoning. *Findings of the Association for Computational Linguistics: EMNLP 2022*.

Paul Röttger, Fabio Pernisi, Bertie Vidgen, and Dirk Hovy. 2025. Safetyprompts: a systematic review of open datasets for evaluating and improving large language model safety. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 27617–27627.

Parker Seegmiller, Joseph Gatto, Omar Sharif, Madhusudan Basak, and Sarah Masud Preum. 2024. Do llms find human answers to fact-driven questions perplexing? a case study on reddit. In *1st Workshop on Reliable Evaluation of Large Language Models for Factual Information (REALInfo-2024), Co-located with AAAI ICWSM'24*.

Parker Seegmiller and Sarah Preum. 2023. Statistical depth for ranking and characterizing transformer-based text embeddings. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9600–9611.

Kaisong Song, Lidong Bing, Wei Gao, Jun Lin, Lujun Zhao, Jiancheng Wang, Changlong Sun, Xiaozhong Liu, and Qiong Zhang. 2019. Using customer service dialogues for satisfaction analysis with context-assisted multiple instance learning. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 198–207.

Dipankar Srirag, Nihar Ranjan Sahoo, and Aditya Joshi. 2025. Evaluating dialect robustness of language models via conversation understanding. In *Proceedings of the Second Workshop on Scaling Up Multilingual & Multi-Cultural Evaluation*, pages 24–38.

Jiuding Sun, Chantal Shaib, and Byron C Wallace. 2024. Evaluating the zero-shot robustness of instruction-tuned language models. In *International Conference on Learning Representations*. ICLR.

Mahan Tafreshipour, Aaron Imani, Eric Huang, Eduardo Santana de Almeida, Thomas Zimmermann, and Iftekhar Ahmed. 2025. Prompting in the wild: An empirical study of prompt evolution in software repositories. In *2025 IEEE/ACM 22nd International Conference on Mining Software Repositories (MSR)*, pages 686–698. IEEE.

Damien Teney, Yong Lin, Seong Joon Oh, and Ehsan Abbasnejad. 2023. Id and ood performance are sometimes inversely correlated on real-world datasets. *Advances in Neural Information Processing Systems*, 36:71703–71722.

Jiayin Wang, Weizhi Ma, Peijie Sun, Min Zhang, and Jian-Yun Nie. 2024a. Understanding user experience in large language model interactions. *arXiv preprint arXiv:2401.08329*.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024b. Improving text embeddings with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11897–11916.

Florian Wenzel, Andrea Dittadi, Peter Gehler, Carl-Johann Simon-Gabriel, Max Horn, Dominik Zietlow, David Kernert, Chris Russell, Thomas Brox, Bernt Schiele, and 1 others. 2022. Assaying out-of-distribution generalization in transfer learning. *Advances in Neural Information Processing Systems*, 35:7181–7198.

Yulong Wu, Viktor Schlegel, and Riza Batista-Navarro. 2025a. Natural context drift undermines the natural language understanding of large language models. *arXiv preprint arXiv:2509.01093*.

Yulong Wu, Viktor Schlegel, and Riza Batista-Navarro. 2025b. Pay attention to real world perturbations! natural robustness evaluation in machine reading comprehension. *arXiv preprint arXiv:2502.16523*.

Zhentao Xu, Mark Jerome Cruz, Matthew Guevara, Tie Wang, Manasi Deshpande, Xiaofeng Wang, and Zheng Li. 2024. Retrieval-augmented generation with knowledge graphs for customer service question answering. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2905–2909.

Linyi Yang, Shuibai Zhang, Libo Qin, Yafu Li, Yidong Wang, Hanmeng Liu, Jindong Wang, Xing Xie, and Yue Zhang. 2023. Glue-x: Evaluating natural language understanding models from an out-of-distribution generalization perspective. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12731–12750.

Yilun Yang and Yekun Chai. 2025. Codemixbench: Evaluating code-mixing capabilities of llms across 18 languages. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 2139–2169.

Rui Ye, Rui Ge, Xinyu Zhu, Jingyi Chai, Du Yaxin, Yang Liu, Yanfeng Wang, and Siheng Chen. 2024. Fedllm-bench: Realistic benchmarks for federated learning of large language models. *Advances in Neural Information Processing Systems*, 37:111106–111130.

Lifan Yuan, Yangyi Chen, Ganqu Cui, Hongcheng Gao, Fangyuan Zou, Xingyi Cheng, Heng Ji, Zhiyuan Liu, and Maosong Sun. 2023. Revisiting out-of-distribution robustness in nlp: Benchmarks, analysis, and llms evaluations. *Advances in Neural Information Processing Systems*, 36:58478–58507.

Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. When scaling meets llm finetuning: The effect of data, model and finetuning method. In *The Twelfth International Conference on Learning Representations*.

Justin Zhao, Timothy Wang, Wael Abid, Geoffrey Angus, Arnav Garg, Jeffery Kinnison, Alex Sherstinsky, Piero Molino, Travis Addair, and Devvret Rishi. 2024a. Lora land: 310 fine-tuned llms that rival gpt-4, a technical report. *arXiv preprint arXiv:2405.00732*.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M Meyer, and Steffen Eger. 2019. Moverscore:

Text generation evaluating with contextualized embeddings and earth mover distance. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics.

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

Yang Zhao, Li Du, Xiao Ding, Kai Xiong, Zhouhao Sun, Shi Jun, Ting Liu, and Bing Qin. 2024c. Deciphering the impact of pretraining data on large language models through machine unlearning. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 9386–9406.

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

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

Yaowei Zheng, Richong Zhang, Junhao Zhang, YeYanhan YeYanhan, and Zheyan Luo. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 400–410.

Ruiqi Zhong, Heng Wang, Dan Klein, and Jacob Steinhardt. 2024. Explaining datasets in words: Statistical models with natural language parameters. *Advances in Neural Information Processing Systems*, 37:79350–79380.

Xinlin Zhuang, Jiahui Peng, Ren Ma, Yinfan Wang, Tianyi Bai, Xingjian Wei, Jiantao Qiu, Chi Zhang, Ying Qian, and Conghui He. 2025. Meta-rater: A multi-dimensional data selection method for pre-training language models. *arXiv preprint arXiv:2504.14194*.

A Complete Dataset Collection Details

To prepare the WildChat dataset for use in our experiments, we begin with the version which has had toxic conversations, and messages containing PII, removed from the dataset[10]. We subset to obtain only the 479k conversations which were classified to be in the English language, and we also take only unique prompts. To ensure that we have {query, response} pairs available for LLM instruction fine-tuning, we subset to consider only the first user prompt and the first response from the OpenAI model. This leaves us with a dataset containing 479k {prompt, model response} pairs which are each associated with the timestamp of the initial message and the hashed IP address of the user. As seen in Figure 5, we then split the dataset across three axes.

Time To bucket the dataset across time, we sort the dataset by timestamp and divide it into 12 equally-sized buckets, which we refer to as "months" for simplicity. To create {ID, OOD} pairs, we are interested in evaluating how well an ID training dataset, comprised of all data up to a given timestamp, performs in an OOD setting defined by a certain "lag." We randomly sample 1k OOD evaluation data from each month $N \in \{1, \ldots, 11\}$. Oracle data for each OOD evaluation setting is a randomly sampled 10k rows from all months up to and including N, i.e. $0 <= Orcl <= N$, ensuring that there is no overlap between OOD eval and oracle training data. This represents the no-lag scenario. ID data is then randomly sampled from each possible collection of months $0 <= M$, where $M < N$. This is the scenario in which some amount of lag is introduced, and enables us to investigate the difference in performance between models trained on $lag = N - M$ data (ID) and no lag data (oracle).

User Group One reason that prompt distributions may change in deployment is the introduction of new users. We treat anonymized IP addresses as proxies for unique users (Ye et al., 2024). As most individual users do not submit a sufficient number of prompts to train a model or analyze performance trends in isolation, we divide users into user groups based on two latent behavioral dimensions: Adoption stage (early, medium, late), determined by when a user first appeared in the dataset; and Query volume (low, medium, high), based on

[10] https://huggingface.co/datasets/allenai/WildChat-1M

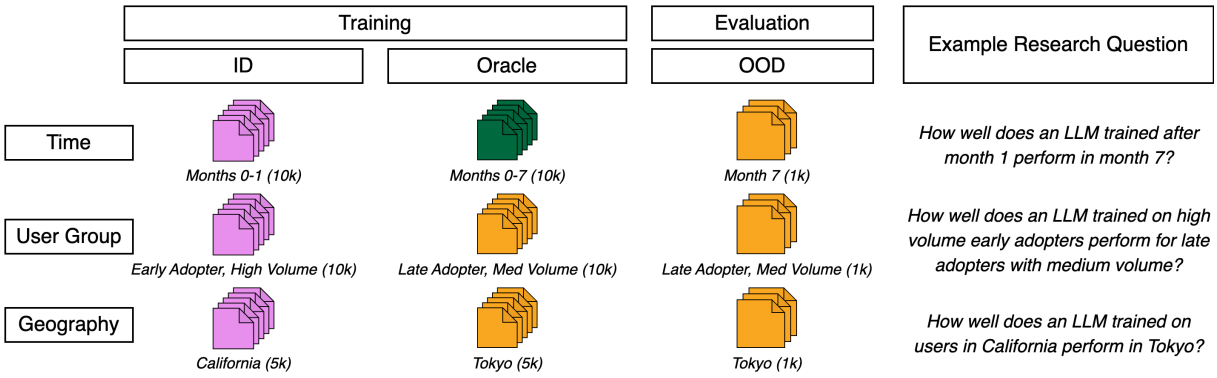

Figure 5: We bucket the WildChat (Zhao et al., 2024b) dataset across three axes to create 192 natural prompt shift settings, training a total of 81 models on 4.68M training instances, and evaluating on 57.6k total evaluation instances. Here we give examples of each natural prompt shift setting, along with motivating research questions for each.

the rate of queries submitted per day. The three buckets for adoption rate and query volume are defined by the 33rd and 66th percentiles. This results in 8 usable user groups (excluding late adopters with low volume, due to insufficient data). For each group, we sample 10k training and 1k evaluation prompts. We consider evaluation scenarios in which oracle training and OOD evaluation data come from the same user group, and ID training data comes from a different group, allowing us to isolate the effect of introducing new user groups on model performance.

Geography WildChat uses IP-to-location mappings to assign each query a geographic region. These regions are recognized at the national and state/city level. We consider city/state-level granularity and select 9 regions with sufficient data coverage (5k for training, 1k for evaluation). These states serve as coarse representations of distinct geographical regions and consist of California, Krasnodar Krai, Massachusetts, Michigan, Moscow, New York, Paris, Pennsylvania, and Tokyo. We consider evaluation scenarios in which oracle training and OOD evaluation data come from the same location, and ID training data comes from a different location, allowing us to assess performance under natural geographic prompt distribution shift.

B Instruction Tuning Details

As highlighted in Section 2.2, we use three pretrained base models in our experiments. We first use Qwen2.5-7B[11] (Qwen et al., 2025), as this model is open-source and has been used for other work investigating instruction fine-tuning (Chen et al., 2024; Dong et al., 2025). To investigate the relationship between model size and model degradation under natural prompt distribution shift, we employ the larger Qwen2.5-14B[12]. Finally, to compare the effects of natural prompt distribution shift on a model from a different model family, we use Llama3-8B[13] (AI@Meta, 2024), which has similarly been used to investigate instruction fine-tuning (Zheng et al., 2024; Li et al., 2024). By fixing a base model in our evaluation, we can consider pretraining data and procedures to be fixed, enabling us to isolate the effects of natural prompt distribution shift on downstream performance.

In each case where a model is trained, we utilize LLaMA-Factory (Zheng et al., 2024)[14] to standardize our training procedure. We train for 1 epoch using a batch size of 4 on a single A100 (80GB) GPU. We estimate each training run requires an average of 2 GPU hours, for a total of 774 GPU hours for model training.

We train using low-rank adaptation (LoRA) (Hu et al., 2022), which has shown to be a performant and efficient fine-tuning strategy (John Schulman, 2025; Zhao et al., 2024a). We configure our LoRA adaptation with a rank of 8 and an alpha scaling factor of 16. We use the AdamW optimizer with weight decay of 0.01, linear learning rate scheduler with warmup over 10% of the training steps, and gradient clipping at a norm of 1.0. We apply mixed precision training using float16 to optimize memory usage and training speed.

[11] Qwen/Qwen2.5-7B

[12] Qwen/Qwen2.5-14B

[13] meta-llama/Meta-Llama-3-8B

[14] https://github.com/hiyouga/LLaMA-Factory

C Detailed Discussion of Distributional Shift Measures in Natural Language

We detail three broad categories of distribution shift measures for natural language, including the four used in our analysis in Section 3 as well as several others which could be used within the LENS framework to analyze different types of natural prompt distribution shift. Many of these measures are designed for purposes other than distributional shift estimation, for example evaluation of open-ended text generation. However, all of these metrics operate only on paired natural language distributions P and Q, thus they may be repurposed to measure the shift between *any* two natural language distributions.

Embeddings-based measures utilize semantically-meaningful multidimensional text representations combined with traditional measures of divergence to measure shifts in language distributions. In order to not rely on the bias of an external embedding model, the LENS framework utilizes the LLM itself to embed user prompts by employing mean pooling of of the hidden state activations from the final layer of the model's encoder (Wang et al., 2024b). This approach captures the model's own semantic representation of the text, aligned with the LLM's internal language processing mechanisms rather than introducing potentially misaligned representations from separate embedding models.

Pillutla et al. (Pillutla et al., 2021) introduce embeddings-based measure **MAUVE** for measuring open-ended text generation, which compares a pair of text distributions by computing information divergences in a quantized embedding space. For a series of mixture distributions $R_\lambda = \lambda P + (1-\lambda)Q$, MAUVE estimates the area under the divergence curve (created by varying $\lambda \in (0,1)$) by quantizing the embedding space of embedding samples for efficient KL divergence calculation. The goal of MAUVE is to simultaneously capture the probability of Type I and Type II errors in machine language generation — thus repurposing MAUVE to measure divergence between prompt distributions $D(P,Q)$ is meant to estimate the overlap in relative frequency of prompts in P and Q. A MAUVE score closer to 0 indicates higher shift between prompt distributions.

Seegmiller et al. (Seegmiller and Preum, 2023) introduce the **Δ-score**, which uses a similarity-based statistical depth $D_\delta(x_i, P)$ to estimate how representative a text $x_i \sim P$ is of the distribution. The Δ-score, $\Delta(P,Q)$, estimates the probability that a randomly selected text y_i from Q will be more representative of distribution P than a randomly-selected text x_i from P, i.e. $\Delta(P,Q) = Pr[D_\delta(X,P) \leq D_\delta(Y,P)]$. The goal of the Δ-score is to capture how well one text distribution represents another. A Δ-score of 0.5 indicates an equal likelihood of representation, and a lower Δ-score indicates higher shift between prompt distributions.

Other embeddings-based measures which may be used within the LENS framework include **Eigen-Divergence** (Abdulaal et al.), which was originally designed for detecting hallucinations by measuring semantic consistency across LLM outputs, and **average minimum distance** (Binu Jose and Das, 2022), which estimates the smallest distance between two sets of text embeddings.

Language model measures utilize a reference language model to estimate token sequence likelihoods, then estimate shift by utilizing token sequence likelihoods.

Perplexity is a fundamental measure in language modeling that quantifies how well a probability distribution predicts a sample. Perplexity evaluates a trained model's ability to predict text sequences by computing the exponential of the average negative log-likelihood of tokens. Formally, perplexity is defined as $PPL_P(Q) = \exp\left(-\frac{1}{n}\sum_{i=1}^{n}\log p_P(y_i|y_{<i})\right)$. Lower perplexity values indicate that the model assigns higher probabilities to the observed sequences, suggesting better predictive performance. Again, in order to not rely on the bias of an external reference model, the LENS framework utilizes the LLM itself, fine-tuned on samples $\{X_i\}_{i=1}^n \sim P$ to estimate perplexity of samples $\{Y_i\}_{i=1}^n \sim Q$. This approach captures how surprising prompts from the Q distribution appear from the perspective of P.

Other language model measures which may be used within the LENS framework include **zero-shot concatenation perplexity** (Lee et al., 2024), which prompts a pre-trained language model with concatenated samples from each distribution and estimates the pre-trained model's expectation of one sample following the other; **KL divergence** and **reverse KL divergence** (Liu et al., 2024; Bu et al., 2016), which measure the information loss when one probability distribution is used to approximate another, with KL divergence penalizing the model

Table 4: We estimate natural prompt distribution shift between all 192 {ID, OOD} settings along with 72 random sampling settings, using four distribution shift measures averaged across three models. We find that prompt distributions shift significantly above average over time, between user groups, and across geographical location. Results are reported as mean (standard deviation). Results for each of the 192 individual splits can be found in csv format in supplementary materials.

Axis	MAUVE	Δ-Score	Perplexity	BLEU
Time	0.5358 (0.17)	0.4396 (0.05)	0.0100 (0.01)	0.2352 (0.18)
User Group	0.6088 (0.19)	0.4680 (0.06)	0.0075 (0.01)	0.4923 (0.24)
Geography	0.0516 (0.11)	0.3722 (0.10)	0.0186 (0.02)	0.1627 (0.24)
Random	0.9618 (0.01)	0.4991 (0.01)	0.0086 (0.01)	0.5689 (0.07)

for placing mass where the reference distribution has little mass, and reverse KL divergence penalizing the model for failing to place mass where the reference distribution has significant mass; and **JSD divergence** (Lu et al., 2020) which provides a symmetric alternative to KL divergence by measuring the average distance from each distribution to their mixture, making it particularly useful for comparing distributions that have minimal overlap.

Statistical measures estimate divergence between prompt distributions based on explicit token-level patterns and frequency distributions.

BLEU, originally developed for machine translation evaluation (Papineni et al., 2002), provides a statistical measure of n-gram overlap between text distributions. While traditionally used to compare candidate translations against reference translations, we repurpose BLEU to quantify the similarity between prompt distributions P and Q by calculating the geometric mean of modified n-gram precision scores across different n-gram lengths. Specifically, BLEU computes the proportion of n-grams in X that appear in Y, with a brevity penalty applied to account for length discrepancies. Higher BLEU scores indicate greater lexical similarity between distributions, capturing surface-level textual patterns rather than deep semantic relationships. This approach offers complementary insights to embedding-based and reference-model-based measures and is particularly sensitive to structural and phrasal overlap. By examining token overlap at various n-gram levels ($n = 1$ to 4), BLEU offers an estimate of lexical divergence between prompt distributions, measuring explicit differences in vocabulary usage, phrase construction, and local contextual patterns.

A common approach to estimating training dataset/evaluation dataset overlap in NLP tasks is to use **n-gram overlap** to determine whether evaluation samples exist in the training data (McCoy et al., 2023; Razeghi et al., 2022; Kandpal et al., 2023; Elazar et al., 2022). This approach is scalable, and provides a straightforward method for detecting exact or near-exact duplicates, though it may fail to identify semantic duplicates or more complex forms of information leakage that don't preserve the same surface-level text patterns.

Other statistical measures include token rank frequency (**Zipf**) (Holtzman et al., 2019), which takes the KL divergence between discrete token rank frequency distributions, a well-known property of human text, and **word mover score** (Zhao et al., 2019), which computes the minimum cost of transforming a text from one distribution to a text in another distribution.

D Towards Identifying Underlying Prompt Shift Trends

The quantitative distribution shift measures reported in Section 3.1 establish that natural prompt distributions shift significantly over time, between user groups, and across geographic regions, but do not illuminate specific changes in the prompts themselves. To motivate future investigation into the underlying drivers of natural prompt shift, we conduct an initial qualitative analysis of prompt pairs sampled from ID and OOD splits along each axis. For each axis, we randomly select 3 {ID, OOD} splits. Within each split, we construct 10 **shift-representative** prompt pairs by sampling source prompts from the 10% highest TTE depth score (Seegmiller and Preum, 2023) (most source-representative) ID prompts and target prompts from the 10% lowest TTE depth (least source-representative) OOD prompts. By sampling ID prompts with high TTE depth (indicating they are highly representative of the source distribution P) and OOD prompts with low TTE depth (indicating they are poorly represented by P and thus most characteristic of the shift in target distribution Q),

Axis	ID Prompt	OOD Prompt
Time	(Month 4) Make up sentences with this word: impartial	(Month 9) As a prompt generator for a generative AI called "Midjourney"... I will give you a concept, and you will provide a detailed prompt ...
Time	(Month 3) Write short 10 tweets: look at this beach right now	(Month 5) [Format your response using markdown. Use headings, subheadings, bullet points, and bold to organize the information] If "Mom 2" is when the first mother dies and the father remarries ...
Time	(Month 3) Describe how DNA methylations can be passed during replication.	(Month 5) Write an article on "Your Partner in Wellness" ... Make sure that you don't follow ai pattern .. it should be as written with an intermediate level writer ... Also add bold and italic.
User Group	(High Volume) give me a response to {message} to send in a discussion, VERY SHORT, CONCISE & CLEAR. ONLY RETURN THE RAW MESSAGE, DO NOT SAY "Hey here is the message you asked"	(Medium Volume) write some eating disorders that will be taught for a nutritional biochemistry course
User Group	(High Volume) give me a response to {message} to send in a discussion, VERY SHORT, CONCISE & CLEAR. ONLY RETURN THE RAW MESSAGE, DO NOT SAY "Hey here is the message you asked"	(Medium Volume) I don't suggest, your words is under arest! ha-ha.
User Group	(High Volume) MS SQL — Procedural Integrity Constraints, Declarative Integrity Constraints, Not Null, Unique, Default and Check constraints, Primary Key and Referential Integrity ...	(Medium Volume) Based on this job description for a PhD position ... write a CV for my past 3 years of work experience with 3–5 bullet points per role ...
Geography	(Pennsylvania) write a script about bobby petrino losing to north dakota state 45-28	(Massachussetts) Make a vividly detailed story taking place in Ancient Rome about a Roman Emperor...
Geography	(Massachusetts) write a comedic and vividly detailed story set in the TV show Z Nation about 10K, before he met Warren's group ...	(Paris) explique de manière claire pour ma documentation ma pipeline ...
Geography	(Pennsylvania) write a script about georgia southern college asun	(Moscow) Alice Flamand is young married socialite in 1973 who is attacked by unknown assailants ...

Table 5: Additional examples of representative prompts pairs along natural prompt distribution shift axes.

each prompt pair is constructed to reflect the contrast between typical in-distribution prompts and prompts most emblematic of the distributional shift. This results in 30 pairs per axis and 90 total. We manually label any observable differences, e.g. in topic, register, linguistic complexity, task type, and specificity. We emphasize that this analysis is exploratory, intended to surface hypotheses and motivate future work rather than draw statistically generalizable conclusions. In addition to our examples and discussion in Table 3 in Section 3.4, we give three representative examples from each axis in Table 5.

E Tabular Results of Prompt Distribution Shift Over Time, Between User Groups, and Across Locations

As discussed in Section 3.1, the LENS framework measures natural prompt distribution shift using four unsupervised measures of distribution shift: MAUVE (lower MAUVE indicates more shift), Δ-score (lower Δ-score indicates more shift), perplexity (higher perplexity indicates more shift), and BLEU (lower BLEU indicates more shift). Here we give aggregated results over all 3 models in each of the 192 {ID, OOD} settings across each axis: time, user group, and geographical location. To compare these estimates against a baseline amount of shift which would be expected in random sampling, we also measure shift between 72 dataset pairs randomly sampled from all English WildChat prompts.

As shown in Figure 3, and in tabular format in Table 4, significant natural prompt distribution shifts are observed across all three axes. The geographical location axis exhibits the strongest shift across all 4 shift measures, for example with a mean MAUVE score of just 0.05 compared to 0.96 baseline random, 0.61 user group, and 0.54 time. This indicates that different geographical behavioral patterns are strongly associated with changes in prompt characteristics. While we subset to only consider English prompts in the WildChat dataset, we hypothesize that part of this measurable difference may be due to different English dialects, a well-known problem for LLMs (Srirag et al., 2025). Natural prompt distribution shifts are also strong between time-shifted datasets, e.g. with an average MAUVE score of 0.54 vs the random base-

Table 6: ID model loss, win, and tie rates against the oracle model, averaged across 192 {ID, OOD} settings and three models for a total of 57.6k OOD evaluation prompts. We find that models trained on prompts from specific user groups and geographic regions perform particularly poorly in OOD settings, recording 87% and 88% loss rates vs the oracle model trained on OOD data (disjoint from evaluation OOD data). Results are reported as mean (standard deviation). Individual performance for each of the 192 splits can be found in csv format in supplementary materials.

Axis	Loss Rate vs Oracle	Win Rate vs Oracle	Tie Rate vs Oracle
Time	0.4435 (0.05)	0.3706 (0.05)	0.1586 (0.04)
User Group	0.8723 (0.04)	0.0180 (0.01)	0.0926 (0.03)
Geography	0.8820 (0.05)	0.0103 (0.01)	0.0884 (0.04)

line MAUVE score of 0.96. While user groups based on adoption rate and usage volume exhibit the least natural prompt distribution shift among the three axes which we evaluate, the average MAUVE (0.61), Δ-score (0.47), and BLEU (0.49) scores in this group still indicate measurable distributional shift compared to baseline (0.96, 0.50, 0.57 average MAUVE, Δ-score, and BLEU scores respectively). These results show that prompt distributions encountered by deployed LLMs can vary substantially depending on when the model is used, who is using it, and where they are located. We give aggregated results in Table 4.

F Tabular Results of LLM Performance Degradation Over Time, Between User Groups, and Across Locations

As discussed in Section 3.2, the LENS framework evaluates LLM OOD performance by comparing trained models against oracle models trained on OOD data. We train and evaluate three pairs of ID/oracle models for each of the 192 {ID, OOD} settings described in Section 2.1. Each evaluation is on 100 OOD evaluation prompts (disjoint from oracle training data) for a total of 57.6k evaluation prompts, generating an ID model response and an oracle model response, then using GPT-4o to judge which model response better follows the user's instructions (Zhao et al., 2024b). See Figure 7 for the full LLM-As-Judge evaluation prompt.

Average model loss, win, and tie rates against the oracle model for each axis can be seen in Figure 6. We find that models trained on prompts from specific user groups and geographic regions perform particularly poorly in OOD settings, each recording 88% loss rate vs the oracle model trained on OOD data. We also find that models trained on time-shifted prompts perform poorly on OOD/"future" prompts, with a 44% loss rate against the oracle model and only a 37% win rate. We give aggregated results in Table 6, and full results across all 192 splits can be found in csv format in supplementary materials.

G User Prompt Shift and Model Degradation Across User Group and Geography Axes

Figure 6 graphically presents selected performance results in the user group and geography settings. In settings where shift is particularly pronounced (low MAUVE), we observe extreme performance drops in OOD settings: ID model loss rate exceeds 88% on average. In the few cases where MAUVE indicates a lower distributional shift, ID models perform slightly better: for example, the model trained on prompts collected from users in Tokyo, which has a low estimated natural prompt distribution shift from the OOD setting of user prompts from California (MAUVE = 0.54), performs relatively well: 77% average loss rate as opposed to the 88% average loss rate.

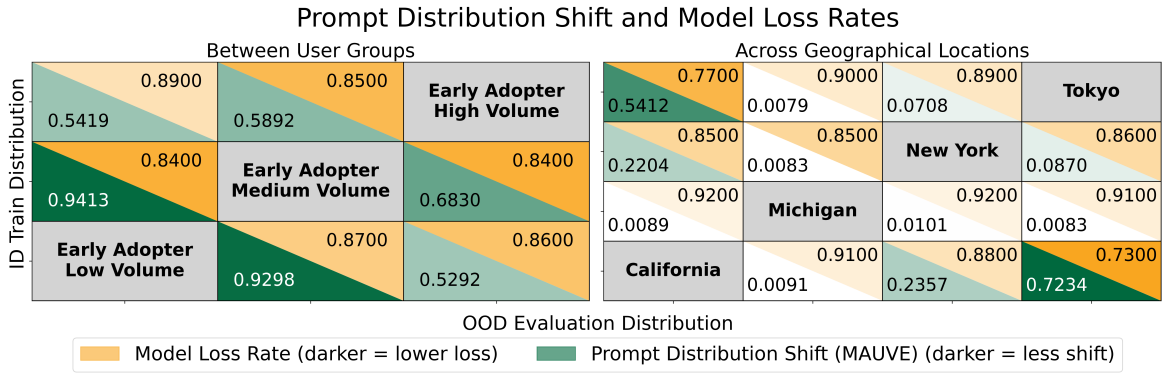

Figure 6: Natural prompt distribution shift (green, with darker meaning less shift) and ID model loss rate (orange, with darker meaning a lower loss rate against the oracle model) are frequently correlated, both between user groups and across geographic regions. For example, users within California and Tokyo exhibit less natural prompt distribution shift than across other regions. Models trained and evaluated on user prompts in these settings also record lower OOD performance loss.

LLM-As-Judge Prompt: Please act as an impartial judge and evaluate the quality of the responses provided by two AI assistants to the user question displayed below. You should choose the assistant that follows the user's instructions and answers the user's question better. Your evaluation should consider factors such as the helpfulness, relevance, accuracy, depth, creativity, and level of detail of their responses. Begin your evaluation by comparing the two responses and provide a short explanation. Avoid any position biases and ensure that the order in which the responses were presented does not influence your decision. Do not allow the length of the responses to influence your evaluation. Do not favor certain names of the assistants. Be as objective as possible. After providing your explanation, output your final verdict by strictly following this format: "[[A]]" if assistant A is better, "[[B]]" if assistant B is better, and "[[C]]" for a tie.
[User Question]
{question}
[The Start of Assistant A's Answer]
{answer_a}
[The End of Assistant A's Answer]
[The Start of Assistant B's Answer]
{answer_b}
[The End of Assistant B's Answer]

Figure 7: Prompt for comparing ID and oracle model responses to OOD evaluation prompts, following WildChat (Zhao et al., 2024b).