Fairness in Prompting: How Demographic Variations Affect Language Model Outputs

Ryandito Diandaru[1] **Gleb Kuzmin**[2,3] **Muhammad Dehan Al Kautsar**[1]
Zeerak Talat[4] **Xudong Han**[1,5] **Haonan Li**[1] **Tatsuki Kuribayashi**[1] **Artem Shelmanov**[1]
Timothy Baldwin[1,5] **Fajri Koto**[1]
[1]MBZUAI [2]FusionBrain Lab [3]RUDN [4]University of Edinburgh [5]LibrAI
{Ryandito.Diandaru, Fajri.Koto}@mbzuai.ac.ae

Abstract

LLMs are expected to respond consistently across demographic groups, yet this assumption remains largely untested due to the absence of demographic information in existing instruction datasets. To address this gap, we introduce `PromptDial`, a collection of 2,289 English prompts written by real users and annotated with seven demographic attributes: sex, race, education, age, language, employment sector, and nationality. We evaluate state-of-the-art generative models on 39 datasets, including machine translation, summarization, grammar correction, knowledge and reasoning, semantics, and question answering, and observe performance disparities of up to 7.7% between demographic groups, with statistically significant differences in over half of the datasets. Our linguistic analysis points to variation in prompt tone and linguistic features as potential drivers of these disparities. Our findings suggest that current instruction tuning practices overlook key aspects of linguistic diversity, and we call for the inclusion of demographic metadata and more representative prompt data to support fairer and more robust language model behavior.[1]

1 Introduction

Instruction tuning transforms generic pre-trained models into responsive assistants capable of interpreting and solving complex tasks effectively (Ouyang et al., 2022; Raffel et al., 2023; Wei et al., 2022). Instruction-tuned large language models (LLMs) have achieved strong results across a wide range of natural language processing (NLP) tasks, including knowledge and reasoning (Cheng et al., 2025; Liu et al., 2024; Bisk et al., 2020; Koto et al., 2023, 2024), and text generation (Venkatraman et al., 2025; Ramprasad et al., 2024). However, instruction tuning alone does not eliminate the social and linguistic biases inherited from pre-training (Rennard et al., 2025; Itzhak et al., 2024). These biases may carry over into inference, where the model's responses can fluctuate as a function of both the task type and the linguistic form of the prompt.

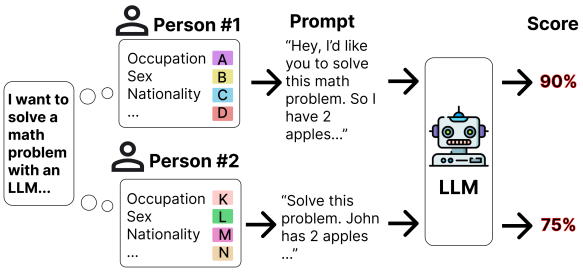


Figure 1: Illustration of how users with different demographic profiles can obtain varying outputs from the same language model when performing the same task.

A critical yet understudied source of such variation is demographic bias: the tendency of a LLM to perform differently, depending on the background of the person providing the prompt. Differences in education, race, age, sex, or language proficiency can influence how users formulate prompts, leading to variation in vocabulary, sentence structure, or spelling (see Figure 1 for illustration). While prior studies have examined model bias in terms of harmful stereotypes (Tomar et al., 2025; Zhang et al., 2024) or downstream disparities (Feng et al., 2023), far less attention has been given to whether LLMs respond equitably to valid prompts phrased in different, demographically grounded styles.

Some work has explored this space through testing prompt variation (Ngweta et al., 2025), role-playing or identity-primed scenarios (Tan and Lee, 2025), but the core question remains open: *does a model's performance remain stable across equally valid prompts that differ only due to the user's demographic profile?* In practice, disparities in this setting could lead to lower-quality responses

[1]Data and code can be accessed at github.com/rayendito/prompt-fairness

Proceedings of the 15th International Joint Conference on Natural Language Processing and the 5th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 106–126
November 6-10, 2026 ©2026 Association for Computational Linguistics

for marginalized communities, even when the intent of the instruction is clear and reasonable. Our work directly addresses this challenge by systematically evaluating the demographic sensitivity of instruction-tuned models across a diverse set of tasks and user groups, providing new insights into the fairness and generalizability of current LLMs. Our contributions can be summarized as follows.

- We introduce `PromptDial`, a collection of 2,289 English prompts written by real users, covering sentiment classification, machine translation, dialogue summarization, topic classification, question answering, free-form text generation, and conditioned text generation. Each prompt is annotated with one of seven coarse-grained demographic attributes: employment sector, race, education, sex, age, language, and nationality group. Unlike previous work that relies on role-playing or synthetic personas, `PromptDial` captures naturally occurring demographically diverse prompting styles.

- We assessed prompt sensitivity using `PromptDial` along with 39 existing datasets, analyzing performance variation between demographic groups. Our evaluation includes diversity scoring and pairwise statistical comparisons between demographic profiles (e.g., Engineering vs. Health, High School vs. Graduate Education). We find that model performance can differ by as much as 7.7% across demographic groups, with over 50% of tasks exhibiting statistically significant differences.

- We conducted a detailed linguistic analysis to investigate potential sources of prompt sensitivity, focusing on variation in prompt tone, grammaticality, and linguistic features, such as the number of short words. Our findings suggest that differences in these features, shaped by users' demographic backgrounds, may contribute to disparities in model performance.

2 Related Work

2.1 Bias in Generative LLMs

Group fairness in LLM is a well-studied (Hardt et al., 2016; Zafar et al., 2017; Cho et al., 2020; Zhao et al., 2020) phenomenon in NLP and is defined as a condition in which an LLM should perform equally well in different demography subgroups (Shen et al., 2022). In other words, LLMs should be insensitive towards prompt variations that may arise from the different backgrounds of their prompters.

2.2 Prompt Sensitivity in LLMs

LLM performance discrepancy can also arise from LLMs which tend to be sensitive towards how the words of the prompt are elicited, i.e. prompt sensitivity. This phenomenon is what causes *prompt engineering*, where many research pursuits try to optimize the performance of LLMs via paraphrasing inputs (White et al., 2023). The rise of this phenomenon proves that there is still efforts to be made after the already-intensive training process of LLMs. Therefore, it is crucial to build LLMs that are robust towards these kind of sensitivities to make LLMs work well and fairly on the same task under many different circumstances regarding the linguistic characteristics of the input.

This prompt sensitivity has been identified in multiple levels by prior works. Zhu et al. (2024) have addressed this problem by introducing perturbations on character, word, sentence, and semantic levels to benchmark how robust LLMs are on these variations. In another work on prompt sensitivity with perturbations, Pezeshkpour and Hruschka (2024) observed a considerable performance gap when LLMs are faced with multiple choice questions whose options have been rearranged.

In their work which has been previously mentioned, Zhu et al. (2024) used Performance Drop Rate (PDR) was used which quantifies relative performance decline for some adversary. In another work, Zhuo et al. (2024) quantified prompt sensitivity of LLMs in their work *ProSa*, in which they introduced *PromptSensiScore* and claimed that higher model decoding confidence correlates with model robustness towards prompt variations. A focused study on quantifying this prompt sensitivity is done by Chatterjee et al. (2024) which introduces `POSIX`, a prompt sensitivity index that tells us how much the model's log-likelihood for a given response shifts when we switch the original prompt for another but intent-preserving one.

Additionally, sensitivity in prompting also exists in Vision Language Models (VLMs). Dumpala et al. (2024) evaluated VLMs using the `SugarCrepe++` dataset to evaluate the robustness of VLMs on lexical alterations and reported that

VLMs are highly sensitive to such modifications. Additionally, Li et al. (2025) also identified this phenomenon in VLMs by introducing *Robust-Prompt* benchmark which then also approached the problem by modeling the variants of prompts to make models more robust.

2.3 Persona-Induced Bias in LLMs

Our work subtly resembles that of Tan and Lee (2025), which views biases purely from the persona of a subject. In their work, scenarios were simulated where there exists a power disparity and whether demographic information about the actors of the scenario was available. The authors then uncover a "default persona" bias favoring middle-aged, able-bodied, native-born, Caucasian, atheistic males with centrist views. They also find that responses involving non-default demographic prompts tend to be lower quality, and that power disparities amplify variability in response semantics and quality.

While their evaluation involves simulated social scenarios, our study focuses on analyzing real-world user prompts in downstream NLP tasks. By linking the prompts to the profiles of their respective authors, we are able to assess whether biases emerge when comparing the prompts authored by different social groups. A key distinction of our approach is that we do not explicitly provide demographic information to the LLMs, allowing us to observe potential biases that arise solely from the interaction between naturally occurring prompts and the models' behavior.

3 Prompt Sensitivity across Different Demography Profiles

3.1 Definitions

Annotator, tasks, and demographic profile. We define a set of demographic profiles, adapted from previous studies (Tan and Lee, 2025; Zhao et al., 2020), *including employment sector, race, education, sex, age, language, and nationality group*. Each attribute has its own set of *labels*, detailed in § 3.1 together with the number of annotators corresponding to each label.[2] We also define a set of tasks, described in Table 2. Under this setup, each annotator provides a set of answers, one per task, subject to the filtering process described in a later section.

[2]We address the imbalance in annotator profiles by limiting the scope of the considered attributes. Further discussion is provided in the *Limitations* section.

Category	Subcategory	Count
Employment Sector	Engineering-STEM	38
	Humanities	35
	Health-STEM	26
Race	White	47
	Asian	26
	Black	18
Education	Postgraduate	42
	Graduate	40
	Highschool	17
Sex	female	52
	male	47
Age	18-30	36
	41-50	32
	31-40	31
Language	L1	43
	L2	20
Nationality Group	Europe	38
	North America	24
	Asia	18
	Africa	15

Table 1: Distribution of respondents based on demography categories and labels.

Fairness. We define a language model to be *fair* on a task t if, in general, the downstream task scores m_1 and m_2 yielded by 2 different prompts p_1 and p_2, where p_1, p_2 come from 2 different annotators with different *labels* from a *demographic profile* on a dataset x, show a negligible performance difference. That is, $m_1 \approx m_2$ given that p_1 and p_2 are *valid* prompts. The *valid* criterion refers to whether a response is reasonable and appropriately addresses the task at hand, as determined by human judgment. We additionally assess *string validity* to ensure that responses can be safely used for string replacement in tasks based on predefined datasets.

3.2 Quantifying Sensitivity and Fairness

Diversity score. To show the distances of task scores between profile labels, we define a *diversity score*. Let $S = (avg_1, avg_2, \ldots, avg_n)$ be an array of averages for each label in a demography profile with each $|S| \geq 2$. A diversity score is calculated for each model against each demography profile on each task group.

$$\text{diversity}(S) = \frac{\sum_{i=1}^{n} \sum_{j=i+1}^{n} |S_i - S_j|}{\binom{n}{2}} \quad (1)$$

Pairwise Welch's t-tests. We also conduct pairwise t-tests on every combination of labels in a

demographic profile and report the percentage of tasks with at least one significantly different pair of labels. We assume that a model is *unfair* on a task for a profile if there is at least one pair of labels within that profile with a p-value that allows us to reject the null hypothesis. Formally, let $T = \{t_1, t_2, ...t_k\}$ be the set of all tasks we have defined and $P_{T,M,D}$ be the percentage of tasks in T inferred with the model M that has at least one pair of labels in the demographic profile D with labels $\{d_1, d_2, ...d_z\}$ that have a p-value of < 0.05. $B_{M,t,d}$ is a list of scores for task t, inferred using model M of the annotators that have the demographic label d.

$$P_{T,M,D} = \frac{|\{t \in T \mid \exists a, b \in D,\ \text{pval}(B_{M,t,a}, B_{M,t,b}) < 0.05\}|}{|T|} \tag{2}$$

Here, $\text{pval}(x, y)$ denotes the p-value obtained from a t-test comparing x and y. We will draw conclusions about the *unfairness* of a model within a demography profile based on these $P_{M,D}$ values. For each of the model disparity measures that we mentioned, we normalize the scores (locally for each model-task combination) prior to doing any calculations involving $B_{M,t,d}$ For example, given the demographic profile D with labels $\{d_1, d_2, d_3\}$, we normalize across $B_{M,t,d_1}, B_{M,t,d_2}, B_{M,t,d_3}$. We do this to ensure comparability between downstream task scores with different ranges.

4 Dataset Construction

We collected the annotator answers through both Prolific (Prolific, 2025) and Google Forms (Google, 2025). We use the former to reach communities where the latter is not widely used. We modified the Potato (Pei et al., 2022) framework for annotator answer collection to fit our use-case. We define 25 tasks that span a diverse range of problems. Each task can have several benchmarks to run on, resulting in 39 task-dataset pairs. The 25 tasks we have defined are further classified into task groups. The complete task definition and classification are presented in Table 2. Annotators are instructed to solve each task as if they are using an LLM. To ensure their understanding, we provide several examples to them prior to the annotation. We deploy surveys iteratively on the Prolific platform, with each run aiming to achieve a background-diverse set of annotators based on our criteria in § 3.1. The examples of different stages of annotation on the annotation platform are presented in Appendix D.

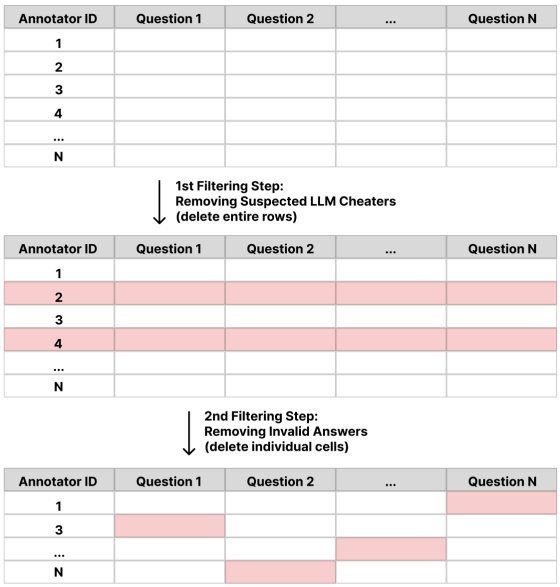


Figure 2: Illustration of the filtering process resulting in a slightly sparse set of annotator responses.

The collected annotator answers go through two filtering phases, as illustrated in Figure 2. First, we exclude obvious *LLM-assisted annotators*: annotators whose responses contain phrases such as "`Here's how you can ask an LLM to solve your task...`". We further identify and discard potential cases of LLM assistance using *LLM detection scores*. Specifically, we compute word-level 3-gram overlaps between each annotator's answers and those produced by several LLMs (`gpt-4o-mini`, `gpt-4o`, and `chatgpt-4o`). For each model, we sample five responses and calculate both the maximum and average similarity scores. An annotator is flagged as LLM-assisted if both the maximum and average scores exceed the thresholds of 0.5 and 0.2, respectively. These thresholds were determined empirically based on the detection scores of annotators whose responses were evidently generated by LLMs prior to this additional verification step. Second, we manually review each annotator's prompts to verify that the corresponding LLM-generated responses are *valid*. In this step, we do not exclude an entire annotator in cases of isolated faulty responses, as it is often the case that the same annotator performs reliably on other tasks. Consequently, the resulting dataset contains a slightly sparse set of responses for each task. After this filtering stage, we retain a total of 99 annotators, yielding a cleaner and more reliable prompt dataset. The final valid answers statistics are presented in Appendix A.

Group	Task Name and Benchmark	Dataset Size	Eval Metric
Semantics	Sentiment Analysis 3 Classes - sentiment_3class (Socher et al., 2013)	500	f1
Semantics	Sentiment Analysis 2 Classes - sentiment_2class (Socher et al., 2013)	500	f1
Machine Translation	Machine Translation (given source and target Languages) (Goyal et al., 2022)	100×14	bleu
Summarization	Dialogue Summarization - dialogsum (Chen et al., 2021)	500	rouge
Semantics	Topic Classification - ag_news (Zhang et al., 2015)	500	f1
Knowledge and Reasoning	Free Generation (Given a Keyword)	1×11	llm_judge
Syntaxes	Free Generation: Grammar Correction	1	llm_judge_grammar_sentence
Syntaxes	Free Generation: Grammar Correction CoT	1	llm_judge
Knowledge and Reasoning	Free Generation: Arithmetic	1	llm_judge
Knowledge and Reasoning	Free Generation: Shakespeare-Prompt Jailbreak	1	llm_judge
QA and Extraction	Multiple Choice Questions - arc_easy (Clark et al., 2018)	500	f1
QA and Extraction	Extractive QA - squad_v2 (Rajpurkar et al., 2018)	500	or_custom
QA and Extraction	MultispanQA - multispan_qa (Li et al., 2022)	500	and_custom
Syntaxes	Grammar Correctness Classification - blimp (Warstadt et al., 2020)	250	f1
Syntaxes	Grammar Correctness Classification - colorless (Gulordava et al., 2018)	250	f1
Conditioned Text Generation	Linguistically-conditioned Text Generation - text_requirement	100	requirement_checker

Table 2: Summary of all tasks and their descriptions.

5 Experiment

5.1 Experimental Setup

Models. We run both closed weight (available through API) and open weight LLMs: `gpt-4o`, `Llama3.1-Instruct (8B, 70B)`, `Llama3.2-Instruct 1B`, `Qwen2.5 Instruct (1B, 7B, 72B)`. For open weight models, we use the `lm-evaluation-harness` (Gao et al., 2024) framework for our evaluation. The metrics used for each task are defined in Table 2; self-defined metrics and llm-as-a-judge prompts are defined in Appendix B. We set the `temperature` and `top_p` the same for every inference, both of which are 1. We use the same evaluation method for the API model, except for classification tasks, for which we employ fuzzy string matching techniques. For tasks with a benchmark, we inject the benchmark dataset instances into the annotator prompts, and for free generation tasks that do not require any dataset, we simply run the annotator prompts. To ensure the applicability and robustness of the llm-as-a-judge evaluation, we compared its annotation with human judgment. We labeled 20% of responses of LLM for two free-form generation tasks and calculated the correlation coefficient and Cohen's κ. As shown in Table 3, both metrics are higher than 0.8, which shows a high level of agreement and justifies the further use of llm-as-a-judge.

Model	Correlation	Cohen's κ
GPT-4o	1.00±0.00	1.00±0.00
Llama3.1-70B-Instruct	0.95±0.05	0.81±0.19
Qwen2.5-72B-Instruct	0.89±0.10	0.89±0.11

Table 3: Pearson's correlation and Cohen's κ between human evaluation scores and llm-as-a-judges scores. Metrics are calculated for a 20% subset of samples for two random tasks with free-form generation, and then averaged.

5.2 Main Results

We begin our analysis by observing Figure 4, which tells us the comparison of unfairness levels for each demography profile for each model. We observed that **nationality group** and **race** almost always have the highest percentage of tasks with a significantly different pair of labels within that demography group. This suggests that these demographic profiles are the ones that the models seem to be most unfair about. At the other end of the spectrum, we see that **sex** has the least percentage of tasks. Further breakdown of the percentage of tasks based on task group is presented in Table 5

On the other hand, Figure 3 tells us how much the differences are in the averages between labels in a demography profile. Despite having some contrasting and consistent demography profiles in Figure 4, we find no pattern that explains the relationship between how many significantly different tasks are in a demography profile and how far are the averages are between each label in them. Further breakdown of the percentage of tasks is presented in Table 5. Further breakdown of the diversity scores based on task groups is presented in Table 4.

To better refine the connection between the scores of the models on the particular task and the diversity scores of the models, we built a scatter plot for all the tasks for all investigated models; the results are provided at Figure 5. The bigger models (GPT-4o, Llama3.1-70B-Instruct, Qwen2.5-72B-Instruct) are mostly grouped in the upper left corner of the plot, showing both high scores on the target

Model	Semantics	Machine Translation	Summarization	Knowledge and Reasoning	Syntaxes	QA and Extraction	Conditioned Text Generation
GPT-4o	1.3%	1.1%	2.1%	3.8%	2.9%	2.0%	0.4%
Llama3.1-70B-Instruct	1.8%	0.8%	1.5%	5.6%	2.7%	3.0%	2.1%
Llama3.1-8B-Instruct	4.2%	0.5%	0.0%	6.1%	6.1%	2.7%	5.1%
Llama3.2-1B-Instruct	3.6%	0.4%	0.0%	6.3%	6.1%	4.4%	2.5%
Qwen2.5-72B-Instruct	1.7%	0.6%	1.8%	5.1%	3.1%	2.6%	1.5%
Qwen2.5-7B-Instruct	3.8%	0.7%	0.0%	6.9%	6.5%	2.3%	4.5%
Qwen2.5-1.5B-Instruct	4.1%	0.4%	0.0%	6.3%	7.7%	3.6%	1.6%

Table 4: Diversity scores of each model against each task group. Each value in this table is the average of all **demography profile**, that is, the average of $P_{M,D}$ across all D. Values are shown in percentage to better highlight their relative magnitudes.

Model	Semantics	Machine Translation	Summarization	Knowledge and Reasoning	Syntaxes	QA and Extraction	Conditioned Text Generation
GPT-4o	66.7%	64.3%	0.0%	61.5%	100.0%	33.3%	100.0%
Llama3.1-70B-Instruct	33.3%	64.3%	0.0%	61.5%	75.0%	66.7%	100.0%
Llama3.1-8B-Instruct	33.3%	71.4%	100.0%	46.2%	100.0%	100.0%	100.0%
Llama3.2-1B-Instruct	33.3%	35.7%	100.0%	30.8%	100.0%	33.3%	100.0%
Qwen2.5-72B-Instruct	66.7%	50.0%	0.0%	30.8%	50.0%	66.7%	0.0%
Qwen2.5-7B-Instruct	100.0%	35.7%	0.0%	46.2%	50.0%	0.0%	100.0%
Qwen2.5-1.5B-Instruct	66.7%	64.3%	0.0%	38.5%	50.0%	66.7%	0.0%

Table 5: Percentage of statistically significant tasks for each model on each task group under **pairwise t-test** with a threshold of $p < 0.05$. For this table, the set of significant tasks considered is the result of all significant tasks over all demography profile D under the union operation, that is, $\bigcup \{t \in T \mid \exists a, b \in D,\ \text{pval}(B_{M,t,a}, B_{M,t,b}) < 0.05\}$. for all D.

task and low diversity scores. On the other hand, smaller models (Llama3.2-1B-Instruct, Qwen2.5-1.5B-Instruct) tend to group in the lower left corner with a low target score and a relatively low diversity score, while medium-sized models are positioned between the two aforementioned model groups. In terms of diversity score, most models on most tasks have a diversity score lower than 10%. However, models of all sizes have a diversity score higher than 10% at least on one task. These results additionally highlight that even large models with high scores on most tasks can have relatively high diversity scores on some tasks. Moreover, these results are supported by Figure 3, where larger models have lower diversity scores than smaller ones, but their mean diversity score is still not equal to zero.

5.3 Prompt Sensitivity Analysis

One of the possible sources of prompt sensitivity is the inherent linguistic characteristics of each prompt. To further investigate this, we extracted several linguistic and LLM features from prompts on each task, namely: (1) number of words, (2) exclamation ratio, (3) type-token ratio (TTR), (4) pronoun ratio, (5) adjective ratio, (6) noun ratio, (7) capitalized words ratio, (8) question ratio, (9) prompt tone, and (10) grammatical correctness. The extraction was conducted using the NLTK framework (Bird and Loper, 2004), with all ratio-based features calculated as a ratio of value to the number of words. The latter two features are categorical and are extracted as follows. Prompt *tone/tonality* classifies each prompt into one of three categories:

1. **Imperative**: A command or instruction (e.g., "Describe the process of photosynthesis.").
2. **Interrogative**: A question (e.g., "What is the process of photosynthesis?").
3. **Declarative**: A statement (e.g., "I'd like to know about the process of photosynthesis.").

We conduct the prompt tone and grammar correctness classification using the `chatgpt-4o-latest`.[3] The exact prompt strings used for classification are in Appendix B. Note that we describe only the features that are used in further analysis and have less than 0.9 pairwise correlation.

[3]We conducted a preliminary experiment with this API model for the tone classification task and report an accuracy of **97%**. However, we do not perform the same verification for grammar correctness, as it has been done extensively (Kobayashi et al., 2024; Davis et al., 2024; Ide et al., 2025).

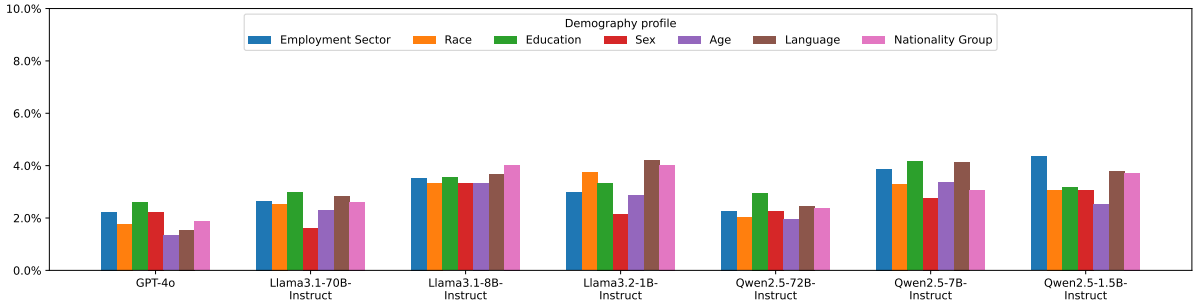


Figure 3: Distribution of **diversity scores** across different task groups for each demography profile. Each bar represents the diversity score for a demography profile on a task group. The diversity scores range from 0 to 1. We limit the y-axis to 10% to better highlight the differences among the values.

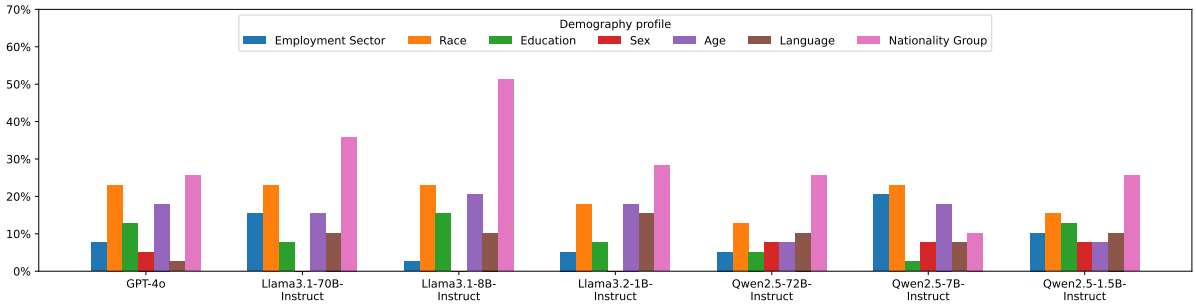


Figure 4: Distribution of the percentage of tasks (from all task groups) where there exists at least one significantly different profile label (Welch's t-test) for each demography profile for each model.

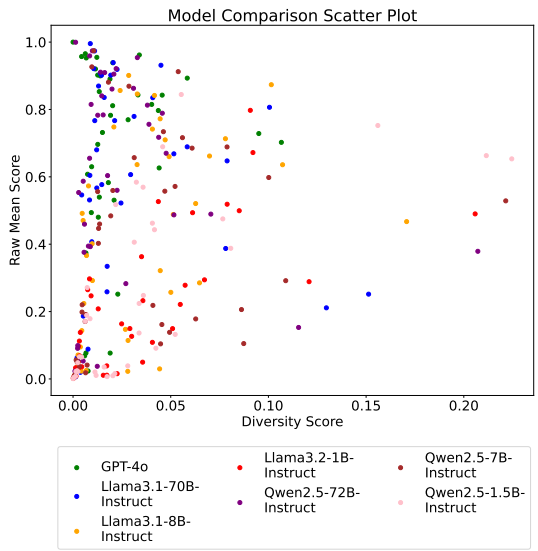


Figure 5: Raw score and diversity score (averaged per demography profile) for each of 39 tasks for each model.

After that, we trained a Linear Support Vector Machine (Pedregosa et al., 2011) for binary classification to predict the demographic profile from these features and used the absolute values of the coefficients of the trained models as an indicator of the importance of each feature. Note that each feature scale is standardized, and the coefficients are comparable across different features. We trained a separate model for each profile. The results are provided in Figure 6 for the top 5 most important features for each demographic characteristic.

Based on the results in Figure 6, we choose the top 5 most common features for all demographic profiles and build a Table 6 showing the percentage of tasks with significantly different scores with respect to these top 5 features. To reduce the number of possible linguistic feature groups, we binarized each linguistic feature category into two groups: less than or equal to the median and greater than the median. The median values are chosen independently for each task.

From Table 6, we highlight TTR, the nouns ratio, the number of words, and the prompt tone for having a noticeable number of different task scores. We observe that various prompt parameters consistently yield significantly different scores compared to other prompt parameters. Particularly, this table suggests that LLMs are less robust towards the aforementioned features than toward the exclamation ratio.

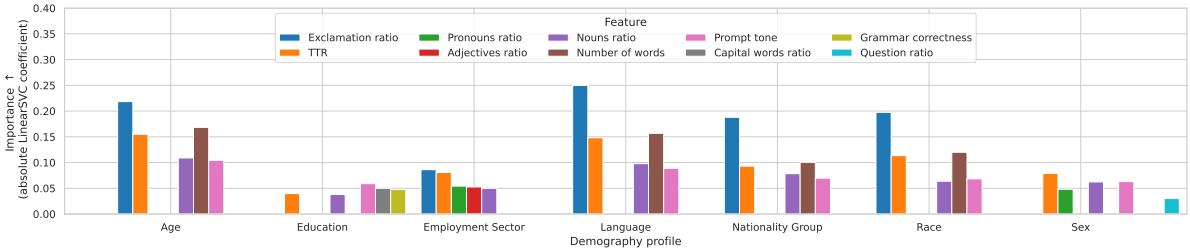


Figure 6: Top-5 features by importance with respect to demography profile.

To further investigate the differences in prompts based on LLM and linguistic features, we built an additional Figure 7 in Appendix C, showing the split of prompts by demographic profile and feature. Figure 7 shows that the distribution for four selected features is more unbalanced between various demographic profiles than for the remaining feature (exclamation ratio), with a lower difference in task scores in Table 6. For example, the ratio of nouns to the number of words differs for individuals with various language profiles, highlighting the possible differences in task scores. The same results apply to race (Asians tend to use more than the median words and nouns ratio), age (the group aged 18 to 30 uses more than the median number of words and nouns), and nationality groups.

Model	Exclamation ratio	TTR	Nouns ratio	Number of words	Prompt tone
GPT-4o	3%	38%	31%	33%	44%
Llama3.1-70B-Instruct	0%	26%	18%	23%	26%
Llama3.1-8B-Instruct	10%	23%	23%	28%	44%
Llama3.2-1B-Instruct	8%	33%	26%	31%	44%
Qwen2.5-72B-Instruct	15%	26%	21%	15%	44%
Qwen2.5-7B-Instruct	10%	13%	28%	23%	36%
Qwen2.5-1.5B-Instruct	10%	31%	13%	15%	33%

Table 6: Percentage of tasks where there exists at least one pair from each features set: *ratio of exclamation*, *TTR*, *ratio of nouns*, *number of words* and *prompt tone* that has a p value of < 0.05. Each set of linguistic features in encoded with respect to median value on the task (e. g. for number of words we compare two groups - less or equal than median and greater than median).

5.4 Discussion

We also conducted additional experiments with other criteria besides pairwise Welch's t-tests, as well as we have more than two labels in demography profiles. Specifically, we employed one-way ANOVA tests and one-way ANOVA tests with Bonferroni correction on the same data as in Table 5; the results are presented in Appendices G and H. However, we noted that the main outcomes, derived from Table 5, remain unchanged; therefore, so we provided the results of the more rigorous tests in the corresponding appendices.

Based on the top-5 features, we hypothesize that the best way to write a safe prompt is to write grammatically correct and detailed prompts; however, we leave the detailed investigation for future work.

6 Conclusion

In this paper, we systematically investigated how demographic variations in prompts influence language model responses. To this end, we introduce a new dataset, `PromptDial`, comprising prompts collected from real users across diverse tasks, where each user is characterized by a unique demographic profile defined by seven attributes.

Using this dataset, we investigated the prompt sensitivity of LLMs with respect to demographic attributes and showed that the performance of LLMs has a statistically significant difference in more than 50% of the investigated tasks, with a maximum difference of 7.7%. Moreover, this disparity in task scores persists across all investigated LLM architectures (Qwen-2.5, Llama-3, and GPT-4o) and for all model sizes (1B to 72B). These results show that even the widely-used modern LLMs can exhibit performance drops based on prompts from a person with a specific demographic profile.

Our additional analysis highlights that several features of prompts can be sources of the described behavior of LLMs, such as prompt tone and linguistic features. We found that the top 5 features affecting the quality of LLM responses' are (1) exclamation ratio, (2) type-token ratio, (3) nouns ratio, (4) number of words, and (5) prompt tone. We showed that the latter four features lead to differences in a task's scores on up to 44% of tasks. Finally, we demonstrated that for some demographic attributes (such as language, age, race, and nationality group), these features have an unbalanced distribution, which, in turn, can lead to score disparity in LLMs.

Limitations

Due to the nonexistence of an annotation platform that *truly* reaches every possible demography label for every demography profile, we limit our findings to only the annotator profiles that we have enumerated and we strive to be as transparent as possible about the distribution of our annotators. We also acknowledge the possibility of one demography label being dominated by another demography label from another demography category e.g.

$$N\% \text{ of } \alpha \text{ are coincidentally } \beta$$

We have made efforts to minimize the effects of this by filtering our annotators when calculating the results. To ensure that our findings are not misleading, we restrict our analysis altogether to *L1 speakers* of English within the following **demographic profiles**: *employment_sector, education, sex* and **labels**: *White (race), Black (race)*. We also limit our study to the English language and tasks that involve only the text modality. We also acknowledge the limitation of using LLM as a judge as the evaluation method of some of our tasks. We chose this approach instead of human evaluation for tasks without automated evaluation, as human assessment is costly and time-consuming.

Our paper also did not investigate why the models evaluated yield these biases in the first place, as they are out of the scope of this study. Regarding this, we suggest a future direction based on our work: investigation of why specific models yield specific types of biases and their mitigation that may be specific to how each models were trained.

Ethical Considerations

We have made sure to adhere to the platform rules for annotator compensation or local norms on how much an annotator's job should be compensated. To protect the anonymity of our annotators, we do not publicly release personally identifiable information (such as names, emails, or phone numbers). We also follow and acknowledge the dataset used in this paper, as cited in Table 2.

We have stated in the annotator registration form that they will agree to be a participant in this research project, which implies that their data will be used to prompt LLMs

References

Steven Bird and Edward Loper. 2004. NLTK: The natural language toolkit. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pages 214–217, Barcelona, Spain. Association for Computational Linguistics.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. 2020. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, pages 7432–7439.

Anwoy Chatterjee, H S V N S Kowndinya Renduchintala, Sumit Bhatia, and Tanmoy Chakraborty. 2024. POSIX: A prompt sensitivity index for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 14550–14565, Miami, Florida, USA. Association for Computational Linguistics.

Yulong Chen, Yang Liu, Liang Chen, and Yue Zhang. 2021. DialogSum: A real-life scenario dialogue summarization dataset. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 5062–5074, Online. Association for Computational Linguistics.

Zhoujun Cheng, Richard Fan, Shibo Hao, Taylor W. Killian, Haonan Li, Suqi Sun, Hector Ren, Alexander Moreno, Daqian Zhang, Tianjun Zhong, Yuxin Xiong, Yuanzhe Hu, Yutao Xie, Xudong Han, Yuqi Wang, Varad Pimpalkhute, Yonghao Zhuang, Aaryamonvikram Singh, Xuezhi Liang, Anze Xie, Jianshu She, Desai Fan, Chengqian Gao, Liqun Ma, Mikhail Yurochkin, John Maggs, Xuezhe Ma, Guowei He, Zhiting Hu, Zhengzhong Liu, and Eric P. Xing. 2025. K2-think: A parameter-efficient reasoning system.

Jaewoong Cho, Gyeongjo Hwang, and Changho Suh. 2020. A fair classifier using kernel density estimation. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv:1803.05457v1*.

Christopher Davis, Andrew Caines, Øistein E. Andersen, Shiva Taslimipoor, Helen Yannakoudakis, Zheng Yuan, Christopher Bryant, Marek Rei, and Paula Buttery. 2024. Prompting open-source and commercial language models for grammatical error correction of English learner text. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11952–11967, Bangkok, Thailand. Association for Computational Linguistics.

Sri Harsha Dumpala, Aman Jaiswal, Chandramouli Sastry, Evangelos Milios, Sageev Oore, and Hassan Sajjad. 2024. Sensitivity of generative vlms to semantically and lexically altered prompts.

Shangbin Feng, Chan Young Park, Yuhan Liu, and Yulia Tsvetkov. 2023. From pretraining data to language models to downstream tasks: Tracking the trails of political biases leading to unfair NLP models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11737–11762, Toronto, Canada. Association for Computational Linguistics.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024. The language model evaluation harness.

Google. 2025. Google forms. Accessed on October 1, 2025.

Naman Goyal, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 evaluation benchmark for low-resource and multilingual machine translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Kristina Gulordava, Piotr Bojanowski, Edouard Grave, Tal Linzen, and Marco Baroni. 2018. Colorless green recurrent networks dream hierarchically. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1195–1205, New Orleans, Louisiana. Association for Computational Linguistics.

Moritz Hardt, Eric Price, and Nathan Srebro. 2016. Equality of opportunity in supervised learning.

Yusuke Ide, Yuto Nishida, Justin Vasselli, Miyu Oba, Yusuke Sakai, Hidetaka Kamigaito, and Taro Watanabe. 2025. How to make the most of LLMs' grammatical knowledge for acceptability judgments. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7416–7432, Albuquerque, New Mexico. Association for Computational Linguistics.

Itay Itzhak, Gabriel Stanovsky, Nir Rosenfeld, and Yonatan Belinkov. 2024. Instructed to bias: Instruction-tuned language models exhibit emergent cognitive bias. *Transactions of the Association for Computational Linguistics*, 12:771–785.

Masamune Kobayashi, Masato Mita, and Mamoru Komachi. 2024. Large language models are state-of-the-art evaluator for grammatical error correction. In *Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2024)*, pages 68–77, Mexico City, Mexico. Association for Computational Linguistics.

Fajri Koto, Nurul Aisyah, Haonan Li, and Timothy Baldwin. 2023. Large language models only pass primary school exams in Indonesia: A comprehensive test on IndoMMLU. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Singapore. Association for Computational Linguistics.

Fajri Koto, Haonan Li, Sara Shatnawi, Jad Doughman, Abdelrahman Sadallah, Aisha Alraeesi, Khalid Almubarak, Zaid Alyafeai, Neha Sengupta, Shady Shehata, Nizar Habash, Preslav Nakov, and Timothy Baldwin. 2024. ArabicMMLU: Assessing massive multitask language understanding in Arabic. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 5622–5640, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Ao Li, Zongfang Liu, Xinhua Li, Jinghui Zhang, Pengwei Wang, and Hu Wang. 2025. Modeling variants of prompts for vision-language models.

Haonan Li, Martin Tomko, Maria Vasardani, and Timothy Baldwin. 2022. Multispanqa: A dataset for multi-span question answering. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1250–1260.

Chen Liu, Fajri Koto, Timothy Baldwin, and Iryna Gurevych. 2024. Are multilingual LLMs culturally-diverse reasoners? an investigation into multicultural proverbs and sayings. 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 2016–2039, Mexico City, Mexico. Association for Computational Linguistics.

Lilian Ngweta, Kiran Kate, Jason Tsay, and Yara Rizk. 2025. Towards LLMs robustness to changes in prompt format styles. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop)*, pages 529–537, Albuquerque, USA. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Jiaxin Pei, Aparna Ananthasubramaniam, Xingyao Wang, Naitian Zhou, Apostolos Dedeloudis, Jackson Sargent, and David Jurgens. 2022. POTATO: The portable text annotation tool. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 327–337, Abu Dhabi, UAE. Association for Computational Linguistics.

Pouya Pezeshkpour and Estevam Hruschka. 2024. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2006–2017, Mexico City, Mexico. Association for Computational Linguistics.

Prolific. 2025. Prolific. Accessed on October 1, 2025.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Sanjana Ramprasad, Elisa Ferracane, and Zachary Lipton. 2024. Analyzing LLM behavior in dialogue summarization: Unveiling circumstantial hallucination trends. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12549–12561, Bangkok, Thailand. Association for Computational Linguistics.

Virgile Rennard, Christos Xypolopoulos, and Michalis Vazirgiannis. 2025. Bias in the mirror : Are LLMs opinions robust to their own adversarial attacks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2128–2143, Vienna, Austria. Association for Computational Linguistics.

Aili Shen, Xudong Han, Trevor Cohn, Timothy Baldwin, and Lea Frermann. 2022. Optimising equal opportunity fairness in model training. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4073–4084, Seattle, United States. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Bryan Chen Zhengyu Tan and Roy Ka-Wei Lee. 2025. Unmasking implicit bias: Evaluating persona-prompted LLM responses in power-disparate social scenarios. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1075–1108, Albuquerque, New Mexico. Association for Computational Linguistics.

Aditya Tomar, Rudra Murthy, and Pushpak Bhattacharyya. 2025. Stereotype detection as a catalyst for enhanced bias detection: A multi-task learning approach. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 17304–17317, Vienna, Austria. Association for Computational Linguistics.

Saranya Venkatraman, Nafis Irtiza Tripto, and Dongwon Lee. 2025. CollabStory: Multi-LLM collaborative story generation and authorship analysis. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 3665–3679, Albuquerque, New Mexico. Association for Computational Linguistics.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners.

Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C. Schmidt. 2023. A prompt pattern catalog to enhance prompt engineering with chatgpt.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P. Gummadi. 2017. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, page 1171–1180, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Proceedings of the 29th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 649–657, Cambridge, MA, USA. MIT Press.

Yunqi Zhang, Songda Li, Chunyuan Deng, Luyi Wang, and Hui Zhao. 2024. Think before you act: A two-stage framework for mitigating gender bias towards vision-language tasks. 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 773–791, Mexico City, Mexico. Association for Computational Linguistics.

Han Zhao, Amanda Coston, Tameem Adel, and Geoffrey J. Gordon. 2020. Conditional learning of fair representations.

Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Yue Zhang, Neil Zhenqiang Gong, and Xing Xie. 2024. Promptrobust: Towards evaluating the robustness of large language models on adversarial prompts.

Jingming Zhuo, Songyang Zhang, Xinyu Fang, Haodong Duan, Dahua Lin, and Kai Chen. 2024. ProSA: Assessing and understanding the prompt sensitivity of LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1950–1976, Miami, Florida, USA. Association for Computational Linguistics.

A Valid Answer Statistics

Task	Valid answers
Sentiment Analysis 3 Classes	80
Sentiment Analysis 2 Classes	90
Machine Translation	92
Dialogue Summarization	97
Topic Classification	90
Free Generation: Balonku Song	93
Free Generation: G30SPKI	93
Free Generation: Hijrah	80
Free Generation: Adhan	84
Free Generation: Karak Kaliang	94
Free Generation: Giant Drum	90
Free Generation: Scientific Letter	83
Free Generation: School Life	97
Free Generation: Olympics Medallists	96
Free Generation: Hillary Clinton	98
Free Generation: MBZUAI	98
Free Generation: Grammar Correction	98
Free Generation: Grammar Correction CoT	98
Free Generation: Arithmetic	99
Free Generation: Shakespeare-Prompt Jailbreak	97
Multiple Choice Questions	88
Extractive QA	91
MultispanQA	92
Grammar Correctness Classification	92
Linguistically-conditioned Text Generation	79

Table 7: Number of valid prompts left after filtering, per task

B Definition of Custom Metrics and Prompts

Custom metrics and exact prompts mentioned throughout the paper are defined here.

Metric Name	**Description or Prompt**
and_custom	Given a model answer and a list of references, get the fuzzy string match score for each reference. Return the average.
or_custom	Given a model answer and a list of references, get the fuzzy string match score for each reference. Return the maximum.
requirement_checker	Given a set of rules, which consists only of minimum generation length and the existence of certain words, score based on how many rules an answer abides by.
llm_judge	You are a judge that gives a score to a prediction answer based on how accurate it is compared to a reference answer. Given this instruction right here: {prompt} On a scale of 0 (worst) to 1 (best), how would you rate the machine's answer? Reference Answer: {refs} Prediction Answer: {preds} Answer only the XML format of the score: <score>[float]</score>

Metric Name	Description or Prompt
llm_judge_drum	You are a judge that gives a holistic score to a text based on how accurate it is to these criteria: - How well it promotes cost - How well it promotes quality - How well it promotes environmental impact This is the text: {preds} Score it on a scale of 0 (worst) to 1 (best). Answer only the XML format of the score: <score>[float]</score>
llm_judge science_project	You are a judge that gives a holistic score to a text based on how accurate it is to these criteria: - How convincing it is This is the text: {preds} Score it on a scale of 0 (worst) to 1 (best). Answer only the XML format of the score: <score>[float]</score>
llm_judge aeni_school_life	You are a judge that gives a score to a text based on how accurate it is to these criteria: 1. The story has to be about: - Aeni's school life 2. The story has exactly: - 2 dogs - 1 cat - 5 mice - 2 adults named 'Thomas' and 'Haroon' - 1 child named 'Aeni' This is the text: {preds} Score it on a scale of 0 (least accurate) to 1 (most accurate). Answer only the XML format of the score: <score>[float]</score>
llm_judge grammar_sentence	You are a judge that gives a score of grammar correctness given a reference and a prediction. On a scale of 0 (worst) to 1 (best), how would you rate the grammar correctness? (including punctuation, etc) Reference: {refs} Prediction: {preds} Answer only the XML format of the score: <score>[float]</score>
prompt tone classifier	Given this request sentence: \"{answer}\"" Classify it as either "Declarative", "Interrogative", or "Imperative". Declarative is a statement-like request. You state a situation that implies a request. Example: I really need some help right now. It would be great if you could stay a bit longer. Interrogative is a question-like request. You ask like a question, but it's really a polite request. Example: Could you help me with this? Would you mind closing the door? Imperative is a command-like request. You give a direct command, often softened with "please." Please pass the salt. Turn down the volume. Answer only in one word.
prompt grammaticality classifier	Given this request sentence: \"{answer}\"" Decide if it's grammatically correct or incorrect. Exclude placeholder strings that are not part of the request. e.g. [TEXT] becuse they are going to be replaced with something else. If it's a request to correct the grammar of some given text, ignore the given text. Only assess the user request string. Answer only in one word. ("correct" or "incorrect")

C Prompt Distribution Based on Demography Profile

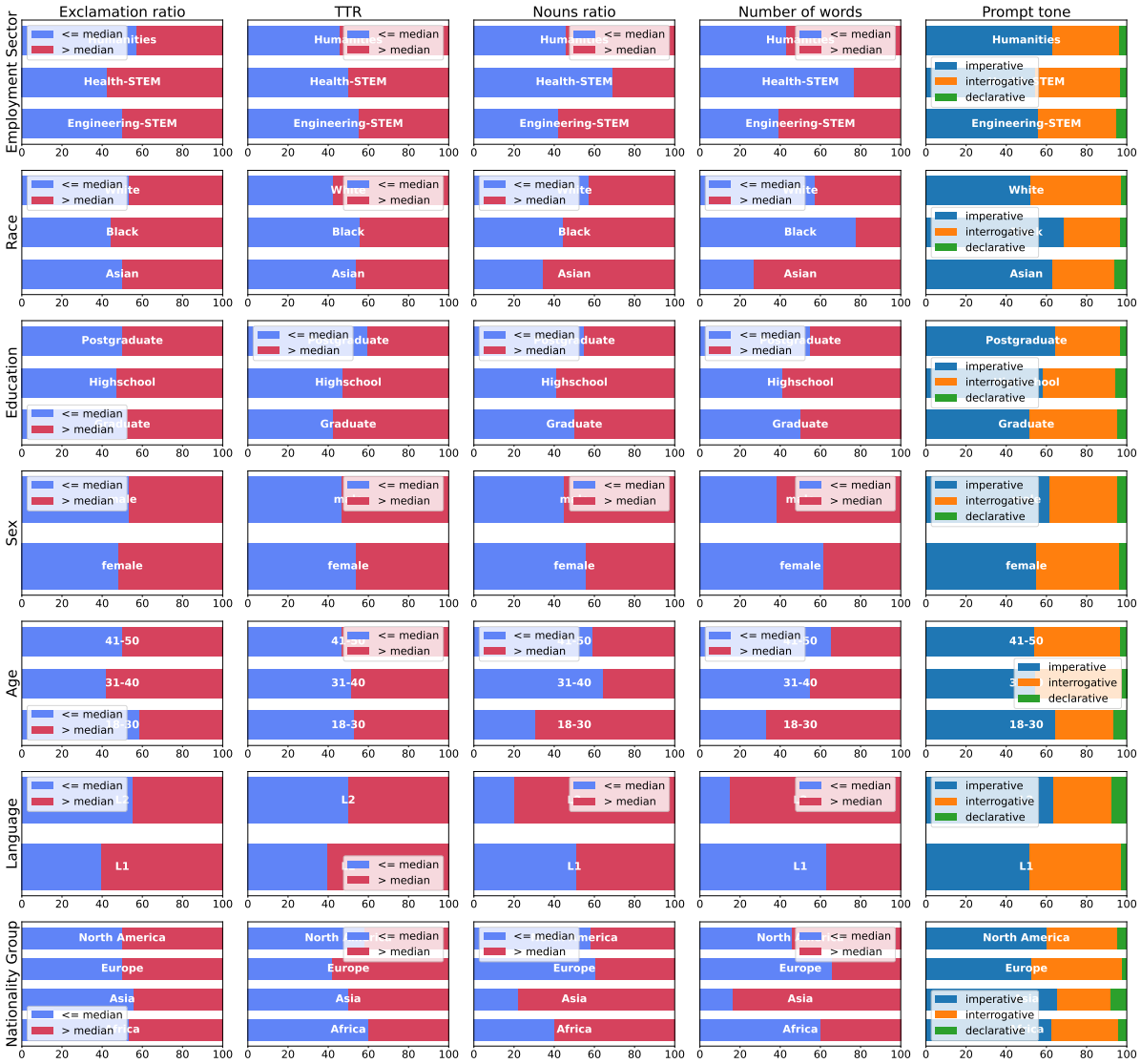


Figure 7: Prompt characteristics across demography profiles for all tasks with respect to linguistic and LLM features.

D Annotation Platform Preview

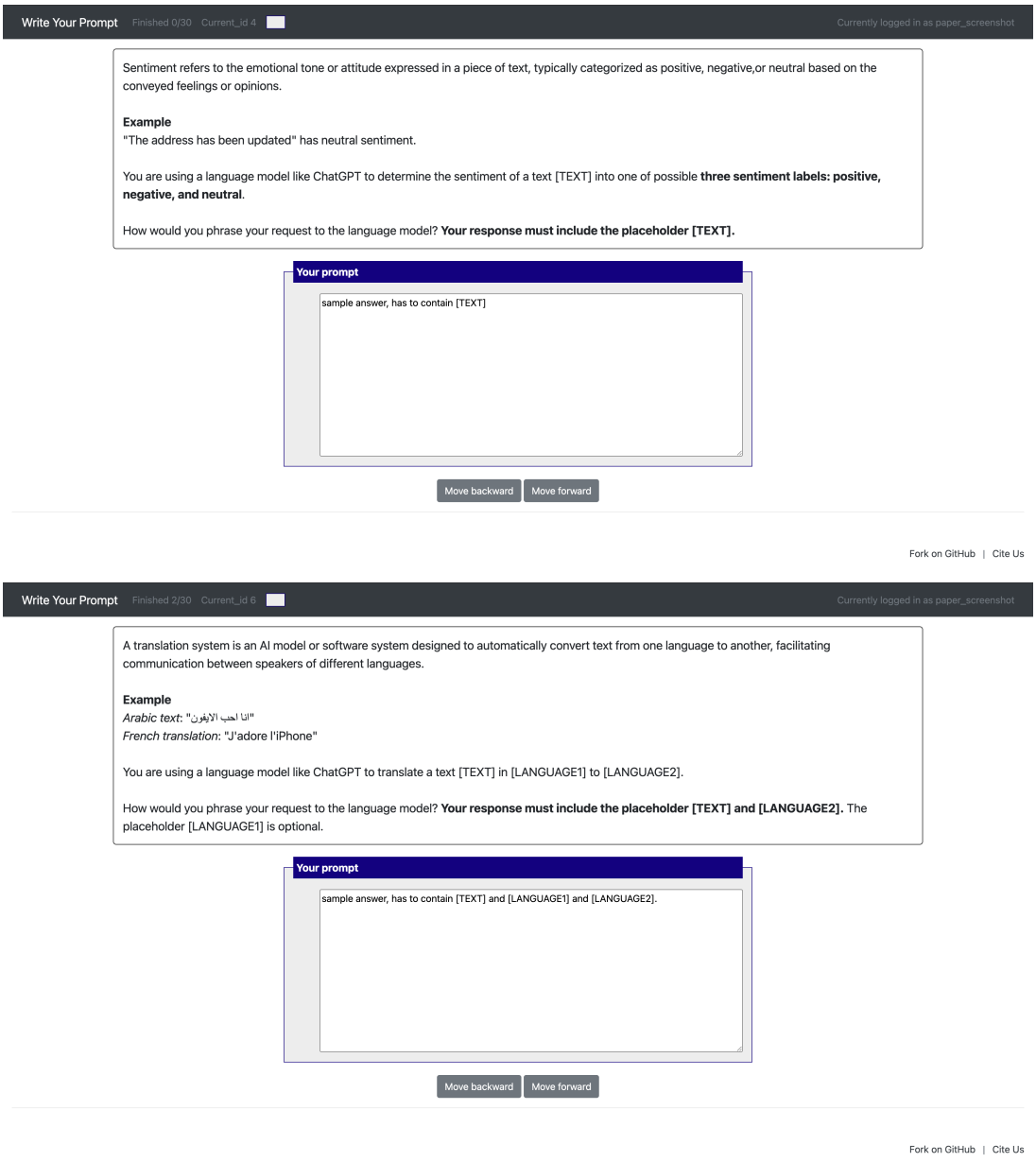
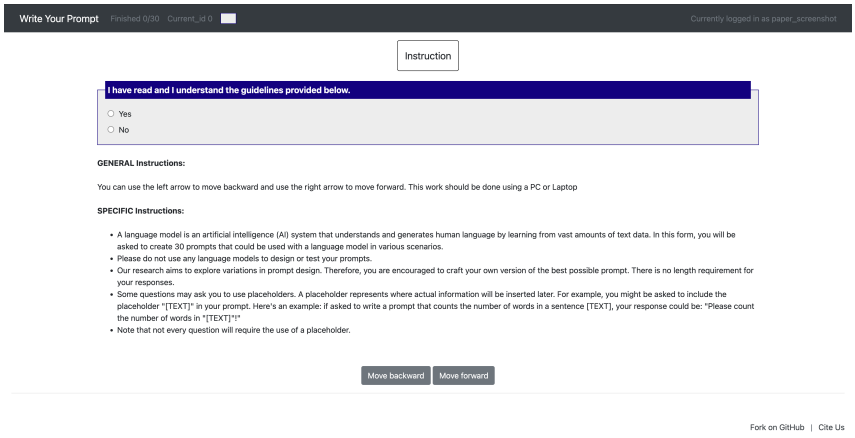
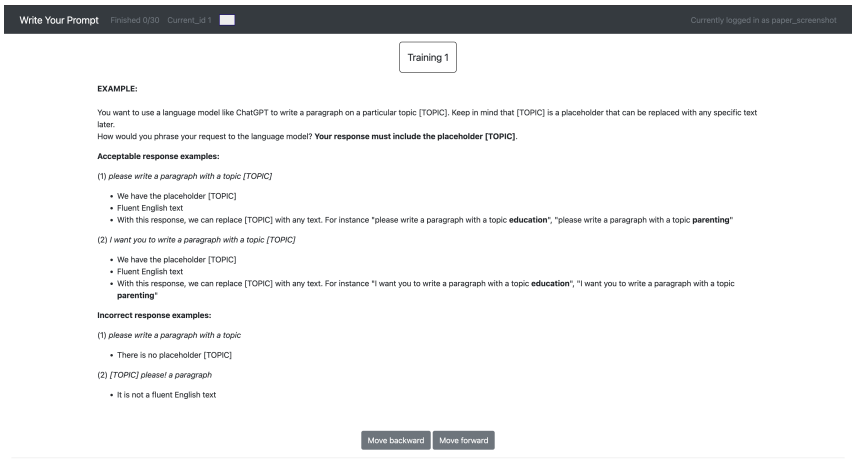


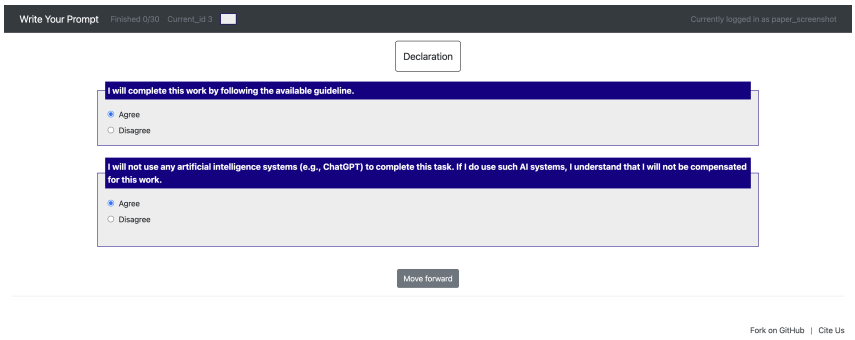
Figure 8: Examples of actual task done by annotators



(a) Introduction page of our data collection platform that contains annotator instructions



(b) First example page of our data collection platform that contains an example of how to do the task



(c) Agreement and declaration of no AI-usage page of our annotation platform

Figure 9: Screenshots of Annotation Platform

E Diversity Scores: Task Group Breakdown

Model	Industry	Race	Education	Sex	Age	Language	Nationality
			Semantics				
GPT-4o	0.08	0.06	0.07	0.06	0.03	0.07	0.04
Llama3.1-70B-Instruct	0.1	0.06	0.06	0.03	0.03	0.07	0.05
Llama3.1-8B-Instruct	0.12	0.05	0.05	0.11	0.03	0.05	0.05
Llama3.2-1B-Instruct	0.1	0.08	0.05	0.04	0.05	0.07	0.07
Qwen2.5-72B-Instruct	0.08	0.05	0.11	0.1	0.04	0.06	0.04
Qwen2.5-7B-Instruct	0.14	0.06	0.09	0.08	0.04	0.07	0.04
Qwen2.5-1.5B-Instruct	0.1	0.07	0.1	0.09	0.07	0.12	0.07
			Syntaxes				
GPT-4o	0.07	0.03	0.11	0.06	0.04	0.02	0.04
Llama3.1-70B-Instruct	0.06	0.06	0.1	0.05	0.04	0.06	0.08
Llama3.1-8B-Instruct	0.06	0.08	0.1	0.02	0.08	0.12	0.12
Llama3.2-1B-Instruct	0.08	0.1	0.09	0.02	0.1	0.13	0.12
Qwen2.5-72B-Instruct	0.08	0.06	0.09	0.08	0.05	0.09	0.08
Qwen2.5-7B-Instruct	0.14	0.1	0.12	0.08	0.13	0.18	0.09
Qwen2.5-1.5B-Instruct	0.15	0.08	0.09	0.11	0.06	0.1	0.11
			Summarization				
GPT-4o	0.05	0.03	0.08	0.04	0.03	0.04	0.04
Llama3.1-70B-Instruct	0.04	0.03	0.06	0.02	0.03	0.02	0.02
Llama3.1-8B-Instruct	0.02	0.05	0.04	0.01	0.05	0.13	0.07
Llama3.2-1B-Instruct	0.06	0.05	0.05	0.01	0.05	0.14	0.1
Qwen2.5-72B-Instruct	0.06	0.02	0.06	0.02	0.03	0.01	0.04
Qwen2.5-7B-Instruct	0.04	0.01	0.06	0.02	0.01	0.04	0.04
Qwen2.5-1.5B-Instruct	0.02	0.01	0.06	0.02	0.01	0.06	0.03
			Knowledge and Reasoning				
GPT-4o	0.08	0.06	0.09	0.05	0.04	0.05	0.05
Llama3.1-70B-Instruct	0.08	0.06	0.11	0.06	0.06	0.08	0.08
Llama3.1-8B-Instruct	0.07	0.05	0.09	0.09	0.06	0.09	0.07
Llama3.2-1B-Instruct	0.08	0.06	0.08	0.06	0.06	0.06	0.07
Qwen2.5-72B-Instruct	0.08	0.05	0.1	0.11	0.05	0.07	0.06
Qwen2.5-7B-Instruct	0.1	0.07	0.08	0.05	0.05	0.07	0.07
Qwen2.5-1.5B-Instruct	0.08	0.05	0.05	0.06	0.04	0.06	0.05
			Machine Translation				
GPT-4o	0.07	0.04	0.07	0.08	0.05	0.03	0.05
Llama3.1-70B-Instruct	0.09	0.06	0.07	0.04	0.04	0.03	0.06
Llama3.1-8B-Instruct	0.07	0.05	0.09	0.06	0.04	0.04	0.07
Llama3.2-1B-Instruct	0.06	0.03	0.08	0.05	0.04	0.03	0.05
Qwen2.5-72B-Instruct	0.09	0.04	0.07	0.05	0.04	0.04	0.06
Qwen2.5-7B-Instruct	0.06	0.03	0.08	0.05	0.04	0.04	0.04
Qwen2.5-1.5B-Instruct	0.08	0.05	0.07	0.06	0.03	0.05	0.06
			Conditioned Text Generation				
GPT-4o	0.05	0.07	0.12	0.12	0.08	0.07	0.1
Llama3.1-70B-Instruct	0.1	0.1	0.1	0.06	0.14	0.13	0.1
Llama3.1-8B-Instruct	0.08	0.05	0.07	0.08	0.11	0.04	0.06
Llama3.2-1B-Instruct	0.02	0.05	0.06	0.1	0.04	0.07	0.07
Qwen2.5-72B-Instruct	0.06	0.08	0.15	0.11	0.07	0.07	0.07
Qwen2.5-7B-Instruct	0.12	0.12	0.18	0.1	0.13	0.14	0.1
Qwen2.5-1.5B-Instruct	0.05	0.03	0.05	0.06	0.03	0.03	0.06
			QA and Extraction				
GPT-4o	0.06	0.04	0.04	0.08	0.03	0.03	0.06
Llama3.1-70B-Instruct	0.07	0.08	0.05	0.04	0.06	0.1	0.07
Llama3.1-8B-Instruct	0.06	0.06	0.09	0.06	0.04	0.06	0.06
Llama3.2-1B-Instruct	0.1	0.11	0.08	0.06	0.04	0.14	0.1
Qwen2.5-72B-Instruct	0.05	0.03	0.08	0.03	0.04	0.04	0.04
Qwen2.5-7B-Instruct	0.02	0.05	0.06	0.04	0.03	0.06	0.04
Qwen2.5-1.5B-Instruct	0.08	0.09	0.07	0.04	0.05	0.07	0.08

Table 8: Breakdown of diversity scores per task group

F Pairwise t-test: Task Group Breakdown

Model	Industry	Race	Education	Sex	Age	Language	Nationality
			QA and Extraction				
GPT-4o	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	33.3%
Llama3.1-70B-Instruct	0.0%	33.3%	0.0%	0.0%	66.7%	33.3%	33.3%
Llama3.1-8B-Instruct	0.0%	66.7%	66.7%	0.0%	33.3%	33.3%	100.0%
Llama3.2-1B-Instruct	0.0%	66.7%	0.0%	0.0%	0.0%	33.3%	66.7%
Qwen2.5-72B-Instruct	0.0%	33.3%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-7B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-1.5B-Instruct	0.0%	66.7%	0.0%	0.0%	0.0%	33.3%	66.7%
			Knowledge and Reasoning				
GPT-4o	7.7%	30.8%	23.1%	0.0%	7.7%	0.0%	23.1%
Llama3.1-70B-Instruct	15.4%	15.4%	7.7%	0.0%	7.7%	0.0%	46.2%
Llama3.1-8B-Instruct	0.0%	15.4%	0.0%	0.0%	15.4%	7.7%	30.8%
Llama3.2-1B-Instruct	15.4%	15.4%	0.0%	0.0%	23.1%	7.7%	15.4%
Qwen2.5-72B-Instruct	0.0%	15.4%	7.7%	15.4%	7.7%	15.4%	15.4%
Qwen2.5-7B-Instruct	23.1%	15.4%	0.0%	7.7%	15.4%	7.7%	23.1%
Qwen2.5-1.5B-Instruct	0.0%	0.0%	15.4%	7.7%	7.7%	0.0%	7.7%
			Semantics				
GPT-4o	0.0%	66.7%	33.3%	0.0%	0.0%	0.0%	33.3%
Llama3.1-70B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	33.3%	33.3%
Llama3.1-8B-Instruct	0.0%	33.3%	33.3%	0.0%	0.0%	0.0%	33.3%
Llama3.2-1B-Instruct	0.0%	0.0%	0.0%	0.0%	33.3%	0.0%	0.0%
Qwen2.5-72B-Instruct	0.0%	0.0%	33.3%	33.3%	0.0%	0.0%	33.3%
Qwen2.5-7B-Instruct	66.7%	33.3%	0.0%	33.3%	33.3%	33.3%	0.0%
Qwen2.5-1.5B-Instruct	0.0%	33.3%	33.3%	33.3%	33.3%	33.3%	33.3%
			Machine Translation				
GPT-4o	14.3%	21.4%	0.0%	14.3%	35.7%	7.1%	14.3%
Llama3.1-70B-Instruct	28.6%	28.6%	7.1%	0.0%	21.4%	0.0%	28.6%
Llama3.1-8B-Instruct	7.1%	14.3%	14.3%	0.0%	14.3%	0.0%	57.1%
Llama3.2-1B-Instruct	0.0%	14.3%	21.4%	0.0%	14.3%	7.1%	21.4%
Qwen2.5-72B-Instruct	14.3%	7.1%	0.0%	0.0%	7.1%	0.0%	28.6%
Qwen2.5-7B-Instruct	14.3%	14.3%	0.0%	7.1%	7.1%	0.0%	7.1%
Qwen2.5-1.5B-Instruct	14.3%	28.6%	7.1%	7.1%	7.1%	14.3%	42.9%
			Conditioned Text Generation				
GPT-4o	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	100.0%
Llama3.1-70B-Instruct	0.0%	100.0%	0.0%	0.0%	100.0%	100.0%	100.0%
Llama3.1-8B-Instruct	0.0%	0.0%	0.0%	0.0%	100.0%	0.0%	0.0%
Llama3.2-1B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	100.0%
Qwen2.5-72B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-7B-Instruct	0.0%	100.0%	100.0%	0.0%	100.0%	0.0%	0.0%
Qwen2.5-1.5B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
			Summarization				
GPT-4o	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Llama3.1-70B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Llama3.1-8B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	100.0%	100.0%
Llama3.2-1B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	100.0%	100.0%
Qwen2.5-72B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-7B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-1.5B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
			Syntaxes				
GPT-4o	0.0%	0.0%	25.0%	0.0%	25.0%	0.0%	50.0%
Llama3.1-70B-Instruct	0.0%	25.0%	25.0%	0.0%	0.0%	25.0%	50.0%
Llama3.1-8B-Instruct	0.0%	25.0%	25.0%	0.0%	50.0%	25.0%	75.0%
Llama3.2-1B-Instruct	0.0%	50.0%	0.0%	0.0%	50.0%	50.0%	75.0%
Qwen2.5-72B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	25.0%	50.0%
Qwen2.5-7B-Instruct	25.0%	50.0%	0.0%	0.0%	50.0%	50.0%	50.0%
Qwen2.5-1.5B-Instruct	50.0%	0.0%	0.0%	0.0%	0.0%	0.0%	25.0%

Table 9: Breakdown of pairwise t-tests per task group

G ANOVA with Bonferroni Correction: Aggregated Results

Model	Semantics	Machine Translation	Summarization	Knowledge and Reasoning	Syntaxes	QA and Extraction	Conditioned Text Generation
GPT-4o	0.0%	14.3%	0.0%	0.0%	0.0%	0.0%	0.0%
Llama3.1-70B-Instruct	0.0%	21.4%	0.0%	23.1%	0.0%	33.3%	100.0%
Llama3.1-8B-Instruct	33.3%	28.6%	100.0%	0.0%	0.0%	33.3%	100.0%
Llama3.2-1B-Instruct	0.0%	7.1%	100.0%	0.0%	25.0%	33.3%	0.0%
Qwen2.5-72B-Instruct	33.3%	0.0%	0.0%	7.7%	0.0%	0.0%	0.0%
Qwen2.5-7B-Instruct	0.0%	7.1%	0.0%	15.4%	50.0%	0.0%	0.0%
Qwen2.5-1.5B-Instruct	33.3%	7.1%	0.0%	7.7%	0.0%	0.0%	0.0%

Table 10: Aggregated scores with ANOVA with Bonferroni Correction for all task groups.

H ANOVA: Aggregated and Task Group Breakdown Results

Model	Semantics	Machine Translation	Summarization	Knowledge and Reasoning	Syntaxes	QA and Extraction	Conditioned Text Generation
GPT-4o	33.3%	42.9%	0.0%	15.4%	0.0%	33.3%	0.0%
Llama3.1-70B-Instruct	33.3%	28.6%	0.0%	38.5%	50.0%	33.3%	100.0%
Llama3.1-8B-Instruct	33.3%	64.3%	100.0%	15.4%	75.0%	66.7%	100.0%
Llama3.2-1B-Instruct	0.0%	14.3%	100.0%	23.1%	75.0%	33.3%	0.0%
Qwen2.5-72B-Instruct	66.7%	7.1%	0.0%	23.1%	50.0%	0.0%	0.0%
Qwen2.5-7B-Instruct	100.0%	28.6%	0.0%	30.8%	50.0%	0.0%	100.0%
Qwen2.5-1.5B-Instruct	33.3%	35.7%	0.0%	23.1%	25.0%	66.7%	0.0%

Table 11: Aggregated scores with ANOVA for all task groups.

Model	Industry	Race	Education	Sex	Age	Language	Nationality
			Summarization				
GPT-4o	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Llama3.1-70B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Llama3.1-8B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	100.0%	100.0%
Llama3.2-1B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	100.0%	100.0%
Qwen2.5-72B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-7B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-1.5B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
			Syntaxes				
GPT-4o	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Llama3.1-70B-Instruct	0.0%	25.0%	0.0%	0.0%	25.0%	0.0%	0.0%
Llama3.1-8B-Instruct	0.0%	0.0%	0.0%	0.0%	25.0%	50.0%	50.0%
Llama3.2-1B-Instruct	0.0%	25.0%	0.0%	0.0%	50.0%	50.0%	50.0%
Qwen2.5-72B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	50.0%	25.0%
Qwen2.5-7B-Instruct	0.0%	0.0%	0.0%	0.0%	50.0%	50.0%	0.0%
Qwen2.5-1.5B-Instruct	25.0%	0.0%	0.0%	0.0%	0.0%	0.0%	25.0%
			Semantics				
GPT-4o	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	33.3%
Llama3.1-70B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	33.3%	0.0%
Llama3.1-8B-Instruct	0.0%	33.3%	0.0%	0.0%	0.0%	0.0%	0.0%
Llama3.2-1B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-72B-Instruct	0.0%	0.0%	33.3%	33.3%	0.0%	0.0%	0.0%
Qwen2.5-7B-Instruct	66.7%	33.3%	0.0%	33.3%	0.0%	33.3%	0.0%
Qwen2.5-1.5B-Instruct	0.0%	33.3%	0.0%	33.3%	33.3%	33.3%	33.3%
			Conditioned Text Generation				
GPT-4o	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Llama3.1-70B-Instruct	0.0%	100.0%	0.0%	0.0%	100.0%	100.0%	100.0%
Llama3.1-8B-Instruct	0.0%	0.0%	0.0%	0.0%	100.0%	0.0%	0.0%
Llama3.2-1B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-72B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-7B-Instruct	0.0%	100.0%	100.0%	0.0%	100.0%	0.0%	0.0%
Qwen2.5-1.5B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
			Machine Translation				
GPT-4o	14.3%	7.1%	0.0%	7.1%	14.3%	0.0%	7.1%
Llama3.1-70B-Instruct	0.0%	14.3%	0.0%	0.0%	0.0%	0.0%	14.3%
Llama3.1-8B-Instruct	7.1%	14.3%	21.4%	0.0%	7.1%	0.0%	35.7%
Llama3.2-1B-Instruct	0.0%	0.0%	7.1%	0.0%	7.1%	0.0%	7.1%
Qwen2.5-72B-Instruct	7.1%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-7B-Instruct	0.0%	0.0%	14.3%	7.1%	0.0%	7.1%	0.0%
Qwen2.5-1.5B-Instruct	0.0%	7.1%	7.1%	7.1%	7.1%	7.1%	14.3%
			QA Extraction				
GPT-4o	0.0%	0.0%	0.0%	33.3%	0.0%	0.0%	0.0%
Llama3.1-70B-Instruct	0.0%	33.3%	0.0%	0.0%	0.0%	33.3%	33.3%
Llama3.1-8B-Instruct	0.0%	0.0%	33.3%	0.0%	33.3%	33.3%	66.7%
Llama3.2-1B-Instruct	0.0%	33.3%	0.0%	0.0%	0.0%	33.3%	33.3%
Qwen2.5-72B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-7B-Instruct	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
Qwen2.5-1.5B-Instruct	0.0%	33.3%	0.0%	0.0%	0.0%	33.3%	33.3%
			Knowledge and Reasoning				
GPT-4o	0.0%	0.0%	7.7%	0.0%	7.7%	0.0%	0.0%
Llama3.1-70B-Instruct	7.7%	15.4%	7.7%	0.0%	7.7%	7.7%	30.8%
Llama3.1-8B-Instruct	0.0%	7.7%	0.0%	7.7%	7.7%	0.0%	0.0%
Llama3.2-1B-Instruct	15.4%	7.7%	0.0%	0.0%	7.7%	7.7%	0.0%
Qwen2.5-72B-Instruct	0.0%	7.7%	0.0%	7.7%	7.7%	15.4%	0.0%
Qwen2.5-7B-Instruct	15.4%	7.7%	0.0%	7.7%	0.0%	15.4%	0.0%
Qwen2.5-1.5B-Instruct	0.0%	0.0%	7.7%	0.0%	7.7%	0.0%	7.7%

Table 12: Breakdown of ANOVA per task group