Assessing Dialect Fairness and Robustness of Large Language Models in Reasoning Tasks

Anonymous ACL submission

Abstract

Language is not monolithic. While benchmarks, including those designed for multiple languages, are often used as proxies to evaluate the performance of Large Language Models (LLMs), they tend to overlook the nuances of within-language variation, and thus fail to model the experience of speakers of non-standard dialects. Focusing on African American Vernacular English (AAVE), we present the first study aimed at objectively assessing the fairness and robustness of LLMs in handling dialects in canonical reasoning tasks, including algorithm, math, logic, and integrated reasoning. We introduce **ReDial** (**Re**asoning with **Dial**ect Queries), a benchmark containing 1.2K+ parallel query pairs in Standardized English and AAVE. We hire AAVE speakers, including experts with computer science backgrounds, to rewrite seven popular benchmarks, such as HumanEval and GSM8K. With ReDial, we evaluate widely used LLMs, including GPT, Claude, Llama, Mistral, and the Phi model families. Our findings reveal that **almost all of these widely used models show significant brittleness and unfairness to queries in AAVE**. Our work establishes a systematic and objective framework for analyzing LLM bias in dialectal queries. Moreover, it highlights how mainstream LLMs provide unfair service to dialect speakers in reasoning tasks, laying a critical foundation for relevant future research.[1]

1 Introduction

Over the last few decades, linguistic research has firmly established that language naturally varies along social, geographic, and demographic dimensions (Chambers and Trudgill, 1998). Such dialectal variation is one of the most salient forms of linguistic diversity. Speakers of "non-standard" dialects are known to experience implicit and explicit discrimination in everyday situations, including housing, education, employment, and the criminal justice system (Baugh, 2005; Adger et al., 2014; Rickford and King, 2016; Drożdżowicz and Peled, 2024). As Large Language Models (LLMs) increasingly serve a broad and rapidly expanding user base (Milmo, 2023; La Malfa et al., 2024), it is critical to understand how they interact with diverse linguistic communities.

In this work, we examine LLMs' **dialect robustness and fairness**. For **robustness**, adversarial robustness provides a consolidated framework to test LLMs on slight variations of existing tasks (Moradi and Samwald, 2021; Jin et al., 2023). Dialects reformulate a problem while maintaining its semantics, i.e., they test what has been referred to as *semantic robustness* (Malfa and Kwiatkowska, 2022). For **fairness**, recent research has demonstrated that LLMs exhibit biases against non-standard dialect queries, predominantly assessed in language and social analysis tasks (Sap et al., 2019; Ziems et al., 2023; Hofmann et al., 2024). Equally relevant, yet less studied, are tasks that require reasoning abilities for problem-solving, decision-making, and critical thinking(Wason, 1972; Huth, 2004; Huang and Chang, 2022; Qiao et al., 2022). For instance, algorithm-related tasks (e.g., generation, debugging, etc.) figure prominently in real user queries, as reflected by their first place on the ArenaHard quality board (Li et al., 2024) and their third place on the WildChat frequency board (Zhao et al., 2024).

However, existing dialectal benchmarks (e.g., Ziems et al., 2023) do not cover these tasks, and current popular reasoning benchmarks such as HumanEval (Chen et al., 2021) and GSM8K (Cobbe et al., 2021) are constructed in Standardized English (SE). This could disadvantage dialect speakers in real-world applications like educational assessment (González-Calatayud et al., 2021), personalized recommendation (Kantharuban et al., 2024), and even multimodal tasks (e.g., voice assis-

[1] Code and data can be accessed here.

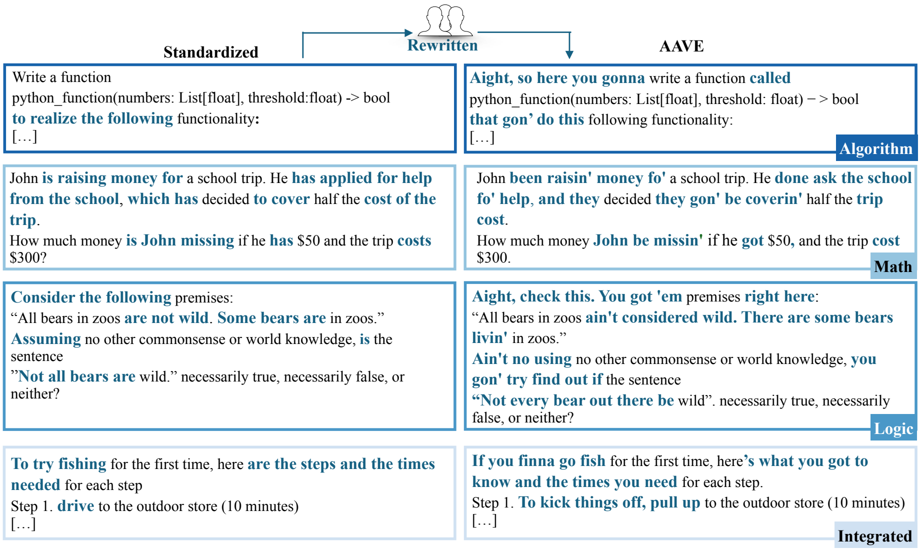

Figure 1: ReDial is a dialect reasoning benchmark composed of 1,200+ SE-AAVE parallel queries. Its source data comes from existing benchmarks in SE. AAVE speakers are hired to rewrite each instance in their dialect but preserve their original intent, meaning, and ground truth output label to form their AAVE counterparts.

tant) (Martin and Wright, 2022), ultimately forcing them to shift their language styles to SE (Cunningham et al., 2024) in order to access the full benefits of modern technologies.

We present the first study that **systematically and objectively evaluates LLM fairness and robustness in reasoning tasks expressed in a non-standard dialect**. We choose AAVE since around 33 million people worldwide and approximately 80% of African Americans in the United States speak AAVE, with reports of discriminative behaviors in various scenarios (Lippi-Green, 1997; Purnell et al., 1999; Massey and Lundy, 2001; Grogger, 2011; Rickford and King, 2016). We hire AAVE speakers to manually rewrite instances from seven popular SE reasoning benchmarks into AAVE (Section 2.1). Our approach has unique advantages compared with prior works which either (i) rely on predefined lexical or morphosyntactic transformation rules (Ziems et al., 2022, 2023), which may overlook subtle contextual nuances, or (ii) use LLM as translators (Gupta et al., 2024), **which may have the very biases that our research wants to unveil** (Fleisig et al., 2024; Smith et al., 2024).

We introduce **ReDial** (Reasoning with Dialect Queries), **the first high-quality, end-to-end human-annotated SE-AAVE parallel dataset for reasoning tasks** (Section 2). ReDial contains over 1.2K SE-AAVE prompt pairs covering four canonical reasoning categories: <u>algorithm</u>, <u>math</u>, <u>logic</u>, and <u>integrated reasoning</u> (tasks that require composing multiple reasoning skills). By anchoring these queries to known correct answers and employing human-based rewriting, ReDial furnishes an objective measure of dialect fairness and robustness. It also avoids the pitfalls of LLM-based evaluations, which can be inherently biased (Zheng et al., 2023; Chen et al., 2024; Shi et al., 2024).

Using ReDial, we benchmark widely used LLMs, including GPT-o1, GPT-4o, Claude-3.5-Sonnet, LLaMA-3.1-70B-Instruct, and others (Section 3). We find that almost all models experience **statistically significant performance drops** on AAVE prompts, despite their semantic equivalence to their SE counterparts. On average, we observe a **relative performance reduction of more than 10%**. This discrepancy persists even with advanced prompting techniques like Chain-of-Thought (CoT) prompting (Kojima et al., 2022; Wei et al., 2022), indicating that current LLMs are both brittle and unfair with dialectal inputs.

To understand these gaps, we further analyze potential causes. Our analysis reveals that the brittleness of LLMs with AAVE prompts arises from a combination of dialect-specific morphosyntactic features and nuanced conversational norms. Experiments with synthetic perturbations and AAVE-specific feature injections show that while these factors contribute to performance degradation, they fail to replicate the severity observed with human-annotated data. This highlights the limitations of rule-based transformations and the critical need for

Category	**Algorithm** (25.7%)		**Logic** (29.8%)		**Math** (24.7%)		**Intergrated** (19.7%)	**Total**
Source	HumanEval	MBPP	LogicBench	Folio	GSM8K	SVAMP	AsyncHow	-
Size	164	150	200	162	150	150	240	1,216

Table 1: ReDial contains $1,216$ fully-annotated parallel prompts for four categories, drawn from seven data sources. Each category contribution to the total amount is reported in percentage points in brackets.

high-quality, context-rich datasets like ReDial to evaluate LLM fairness and robustness effectively. In summary, in this paper:

1. We introduce ReDial, the first high-quality, end-to-end human-annotated AAVE-SE parallel reasoning benchmark spanning four foundational reasoning tasks.
2. We show that leading LLMs exhibit significant unfairness and brittleness on AAVE prompts compared to their SE counterparts.
3. We identify that the brittleness of LLMs with AAVE prompts stems from a combination of dialect-specific morphosyntactic features and nuanced conversational norms, which cannot be captured by synthetic transformations.

2 Dataset

ReDial (**Re**asoning with **Dial**ect Queries) is a benchmark of more than 1.2K parallel Standard English – African American Vernacular English (SE-AAVE) query pairs. Table 1 provides an overview of the distribution, and Figure 1 along with Appendix A.2 present illustrative examples.

Following Zhu et al. (2023a), ReDial includes four canonical reasoning tasks—**algorithm**, **logic**, **math**. We additionally consider **integrated reasoning** as a compositional task requiring multiple skills. The task formulation of ReDial is linguistically diverse, addresses cornerstone problems in human reasoning, and is of particular interest as it is challenging for LLMs.

We first describe the data sources and sampling strategies (Section 2.1), and then detail the AAVE rewriting and validation processes that ensure high data quality (Section 2.2).

2.1 Data Sourcing

We construct a highly curated dataset by drawing upon seven established benchmarks covering different aspects of reasoning. For each source, we provide key references, task descriptions, and sample sizes. Additional examples can be found in Appendix A.1.

Algorithm **HumanEval** (Chen et al., 2021) consists of 164 human-written code completion instances. We convert and include all these code completion headings into instruction-following natural language queries following the paradigm of InstructHumanEval.[2]

Algorithm **MBPP** (Austin et al., 2021) includes $1,000$ code generation queries. We randomly sample 150 instances from its sanitized test set (Liu et al., 2023).

Math **GSM8K** (Cobbe et al., 2021) is a graduate-level math word problem dataset containing $8,790$ instances. We randomly sample 150 instances from its test set.

Math **SVAMP** (Patel et al., 2021) is a collection of $1,000$ elementary-school math problems. We randomly sample 150 instances from its test set.

Logic **LogicBench** (Parmar et al., 2024) comprises various logic questions in both binary classification and multiple-choice formats. We sample 100 binary and 100 multiple-choice questions, collecting 200 samples in total.

Logic **Folio (original+perturbed)** (Han et al., 2022; Wu et al., 2023) Original Folio is a manually curated logic benchmark. We select 81 instances along with their manually perturbed versions from Wu et al. (2023), yielding 162 instances.

Intergrated **AsyncHow** (Lin et al., 2024) is a planning reasoning benchmark. LLMs must interpret natural language descriptions (i.e., logic), find different possible paths in the graph (i.e., algorithm), and then calculate and compare the time cost for these paths (i.e., math) to reach the correct answer. We use stratified sampling based on the dataset's complexity metrics and include 240 instances.

2.2 Annotation and Quality Assurance

After data sourcing, we hire AAVE speakers to rewrite each instance in AAVE. We schematize our annotation and validation in Figure 2 and describe them below.

[2] https://huggingface.co/datasets/codeparrot/instructhumaneval

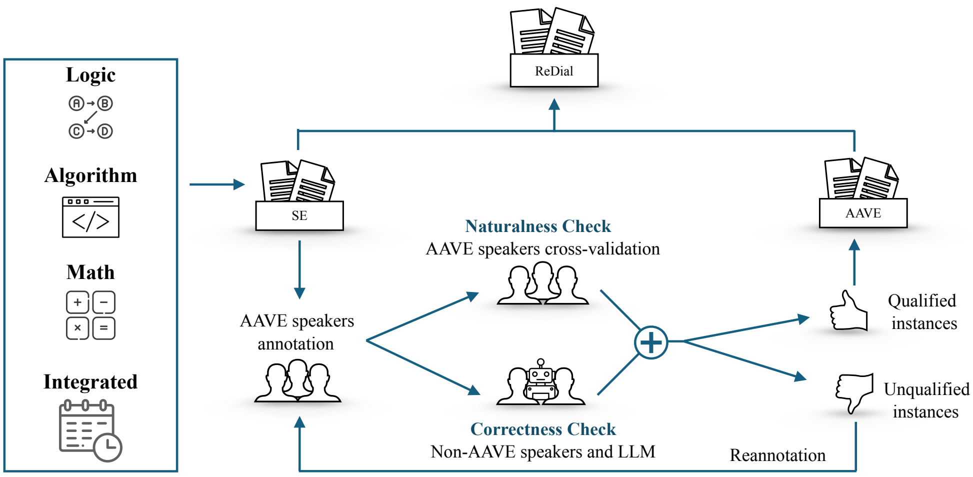

Figure 2: Annotation and cross-validation of ReDial instances. We first sample instances from datasets of four canonical reasoning tasks to compose the source data, then we hire AAVE speakers to rewrite the instances in their dialect. To ensure the quality, we conduct **naturalness check** by AAVE speakers and **correctness check** by non-AAVE speakers and LLMs. We reannotate instances that do not pass the quality checks and iterate the process until the data meet our criteria. Finally, we combine the source data and AAVE rewriting to obtain ReDial.

Annotation. We hire and instruct AAVE speakers to rewrite each SE query so that it sounds natural to AAVE speakers while retaining all critical information (e.g., numerical values, logical conditions, and technical details). For algorithm-related tasks involving code, we hire annotators with a background in computer science to keep the logic and semantics of the code tasks.[3]

Validation. We then perform a careful quality check to ensure both naturalness and correctness. First, we ask annotators to cross-check and edit each others' annotations to make sure that the annotations are **natural** to AAVE speakers. Second, to ensure **correctness**, we first have non-AAVE speakers manually check the essential information, then conduct a sanity check with GPT-4o for the correctness of rewriting (details in Appendix A.4). We **manually check** data that GPT-4o flags as invalid to see if all essential information is preserved. **No instance is rejected solely based on the LLM's judgment.** We return invalid instances to AAVE speakers for correction and iterate the process until all data pass the check.

After these efforts, we obtain ReDial: a high-quality, end-to-end human-annotated SE–AAVE parallel dataset comprising over 1.2K instances spanning four canonical reasoning tasks. In the rest of this paper, we refer to the SE portion of the dataset as *SE ReDial* and the AAVE portion as *AAVE ReDial*.

3 Experiments

3.1 Experimental Setting

3.1.1 Models

We test five families of models, two proprietary and three open-source. The rationale is to benchmark widely used LLMs with impressive reasoning performance. All experiments were conducted between September and December 2024.

GPT. We use GPT-o1 (OpenAI), GPT-4o (OpenAI, 2024), GPT-4 (Achiam et al., 2023), GPT-3.5-turbo (Achiam et al., 2023),[4] as a family of closed-source models to compare with open-source models for dialect robustness. In particular, o1 is trained using large-scale reinforcement learning (RL) to reason through a chain of thoughts and scales inference time computation to achieve highly complex reasoning paths, demonstrating significant improvements in reasoning tasks (OpenAI). We use GPT-o1 model to understand how RL reasoning post-training affects LLMs' dialect robustness and fairness.

Claude. Developed by Anthropic, the Claude 3 model family represents a widely-used proprietary LLM. For our experiments, we utilize the Claude 3.5 Sonnet model (Anthropic).

[3]Details on annotator compensation, qualifications, and guidelines are presented in Appendix A.3.

[4]Proprietary model API version information: o1: GPT-o1-preview; gpt-4o: 2024-05-13; gpt-4: 2024-05-03; gpt-3.5-turbo: 2023-11-06.

LLaMA. We use LLaMA-3-8B / 70B-Instruct and LLaMA-3.1-70B-instruct (Dubey et al., 2024) which are reported for comparable performance with proprietary GPT models.

Mistral / Mixtral. We use Mistral-7B-Instruct-v0.3 (Jiang et al., 2023) and Mixtral-8x7B-Instruct-v0.1 (Jiang et al., 2024). Mistral-7B-Instruct-v0.3 is reported to be outstanding in reasoning; with Mixtral-8x7B-Instruct-v0.1, we can understand whether Mixture-of-Expert architectures enhance dialect robustness.

Phi. We use Phi-3-Mini / Small / Medium-128K-Instruct (Abdin et al., 2024; Gunasekar et al., 2023) in our experiment. Phi-3 models, pre-trained on carefully designed "textbook" data, are reported for impressive performance in reasoning despite their small sizes (3.8/7/14B parameters each). We use these models to understand how highly curated pre-training data affect LLMs' dialect robustness and fairness.

3.1.2 Implementation and Evaluation

Implementation. We set the temperature to zero for the main experiments to ensure maximum reproducibility. We report two prompting methods in our main results: (i) direct prompting LLMs with task instances, which resembles general real-life use cases the most (Direct) and (ii) zero-shot Chain of Thought (Wei et al., 2022; Kojima et al., 2022) (CoT, i.e., adding instructions in the spirit of "`Let's think step by step`" on top of task descriptions, which resembles expert user prompts to improve model performance). For GPT-o1, we only test the direct prompting as its inherent cot reasoning pattern.[5] We report further implementation details in Appendix A.5.[6]

Evaluation. To unify evaluation metrics, we consider the pass rate for all tasks. For Algorithm, we consider Pass@1 using all base and extra unit test cases in EvalPlus (Liu et al., 2023), which results in either pass or fail for every code generation. We convert all other task measures of correctness or incorrectness to pass or fail.

3.2 Experimental Results

We report pass rates for ReDial in Table 2 and 3, and summarize the main results of our experiments.

All Models are Brittle. **All models experience performance drops** in AAVE compared to SE ReDial, and these drops are statistically significant in all cases, with the sole exception of LLaMA-3-8B-Instruct. This indicates that our benchmark poses huge challenges to models, both in terms of absolute performance and with respect to their dialect robustness and fairness.

The absolute performance gaps commonly range from around 5% to over 10% (Δ in table 2). Specifically, GPT-4o (zero-shot) shows an absolute gap of 11.6%, dropping from an average of 0.832 to 0.716. GPT-4 (CoT) exhibits an 11.5% drop. Mixtral-8x7B-Instruct-v0.1 (zero-shot) shows a particularly large difference of 11.4% points as well. Interestingly, we found that although the performance drop of GPT-o1 is smaller than other GPT models, but still significant. It indicates that although further RL post-training on general reasoning and inference scaling can systematically enhance dialect robustness and fairness, they cannot completely solve the problem.

In short, dialect unfairness and brittleness are identified in all the models we examined, including the mixture of expert and large reasoning models. This finding indicates that the problem is widespread, non-trivial, and cannot be easily mitigated by naively changing model architecture or proposing more complex reasoning paths.

All Tasks are Brittle. When aggregated by task type, AAVE queries cause a statistically significant performance drop across all these categories (Table 3). For instance, when averaging results across all models: direct prompting leads to an average 10% relative performance drop.

Interestingly, integrated reasoning tasks, which require multiple reasoning skill compositions, show some of the largest relative drops (about 30%). This suggests that **compositionally complex task may be more prone to dialect brittleness**.

Prompting and Inference Scaling are Brittle. While CoT prompting can slightly reduce the discrepancy for some models, it fails to close it entirely. For example, GPT-4o's performance gap

[5] We also test non-zero temperatures and report results in Appendix A.6.

[6] We deliberately avoid testing advanced prompting methods, such as Tree of Thought (Yao et al., 2024) and Self-Refine (Madaan et al., 2024). Our focus is on evaluating **how LLMs perform when prompted for everyday use by dialect users**, which is critical for assessing fairness in LLMs. Similarly, we do not fine-tune any models, as our study aims to investigate biases inherent in base models. The effects of fine-tuning are beyond the scope of this study.

Model	Setting	Algorithm		Logic		Math		Integrated		All		
		SE	AAVE	SE	AAVE	SE	AAVE	SE	AAVE	SE	AAVE	Δ
GPT-o1	Direct	0.818	0.825	0.947	0.923	0.878	0.815	0.942	0.925	0.892	0.866	**-0.026**
GPT-4o	Direct	0.790	0.761	0.933	0.930	0.818	0.768	0.783	0.312	0.832	0.716	**-0.116**
	CoT	0.771	0.761	0.950	0.920	0.815	0.771	0.762	0.662	0.826	0.784	**-0.043**
GPT-4	Direct	0.742	0.723	0.840	0.713	0.796	0.749	0.217	0.133	0.678	0.612	**-0.067**
	CoT	0.723	0.608	0.920	0.813	0.793	0.743	0.283	0.058	0.706	0.590	**-0.115**
GPT-3.5-turbo	Direct	0.653	0.631	0.667	0.443	0.533	0.544	0.200	0.129	0.531	0.460	**-0.072**
	CoT	0.646	0.551	0.753	0.543	0.503	0.425	0.075	0.067	0.517	0.416	**-0.101**
Claude-3.5-Sonnet	Direct	0.771	0.806	0.970	0.930	0.851	0.776	0.879	0.717	0.865	0.810	**-0.055**
	CoT	0.774	0.736	0.953	0.940	0.859	0.796	0.900	0.771	0.868	0.811	**-0.058**
LLaMA-3.1-70B	Direct	0.726	0.653	0.767	0.893	0.702	0.630	0.392	0.112	0.663	0.599	**-0.064**
	CoT	0.723	0.653	0.880	0.870	0.809	0.768	0.579	0.500	0.759	0.711	**-0.049**
LLaMA-3-70B	Direct	0.682	0.643	0.907	0.887	0.663	0.552	0.158	0.067	0.628	0.562	**-0.066**
	CoT	0.697	0.646	0.923	0.887	0.616	0.561	0.517	0.350	0.693	0.622	**-0.072**
LLaMA-3-8B	Direct	0.535	0.510	0.827	0.800	0.478	0.464	0.025	0.067	0.489	0.480	-0.009
	CoT	0.532	0.478	0.827	0.800	0.475	0.492	0.029	0.025	0.488	0.472	-0.016
Mixtral-8x7B	Direct	0.452	0.401	0.520	0.340	0.414	0.240	0.100	0.075	0.388	0.274	**-0.114**
	CoT	0.468	0.411	0.687	0.567	0.384	0.285	0.133	0.071	0.431	0.345	**-0.086**
Mistral-7B	Direct	0.331	0.255	0.400	0.213	0.315	0.271	0.096	0.075	0.297	0.214	**-0.083**
	CoT	0.312	0.245	0.453	0.347	0.323	0.293	0.083	0.083	0.305	0.252	**-0.053**
Phi-3-Medium	Direct	0.545	0.433	0.867	0.790	0.500	0.470	0.050	0.038	0.513	0.454	**-0.059**
	CoT	0.548	0.455	0.860	0.827	0.492	0.439	0.067	0.029	0.513	0.458	**-0.055**
Phi-3-Small	Direct	0.615	0.252	0.820	0.760	0.530	0.525	0.058	0.062	0.530	0.421	**-0.109**
	CoT	0.570	0.194	0.893	0.843	0.544	0.522	0.096	0.079	0.549	0.429	**-0.119**
Phi-3-Mini	Direct	0.557	0.427	0.520	0.550	0.605	0.525	0.021	0.042	0.456	0.410	**-0.046**
	CoT	0.576	0.443	0.773	0.750	0.622	0.528	0.017	0.021	0.528	0.461	**-0.067**

Table 2: We report model pass rates using direct and CoT prompting on ReDial, including individual performances on subtasks and overall performance/gap (in column **all**). We follow the recommendations from (Dror et al., 2018) and test the statistical significance of performance differences between SE and AAVE on **all** results using the McNemar's test for binary data (McNemar, 1947). We correct p-values for multiple measurements using the Holm-Bonferroni method (Holm, 1979). Statistically significant drops are in **bold**. Details in Appendix A.7.

	Algorithm	Math	Logic	Intergrated	All
SE	0.632	0.622	0.768	0.302	0.597
AAVE	0.563	0.564	0.706	0.212	0.529
Δ	**-0.069**	**-0.058**	**-0.062**	**-0.090**	**-0.068**

Table 3: Pass rates by task averaged across responses from all models with direct prompting. In **bold**, results statistically deviate according to McNemar's tests applied to AAVE and SE (i.e., models have significant drops in AAVE). We also report the SAE-AAVE absolute delta in performance.

decreases from about 0.116 (zero-shot) to 0.055 (CoT). This suggests that **even when models are given additional reasoning "scaffolding," their understanding and performance in AAVE remain comparatively weaker than in SE**, which is also in line with our observation with GPT-o1 results.

Model Scaling is Brittle. All model families display some degree of dialect-related performance degradation. A notable observation is that **simply using larger models does not inherently improve robustness to AAVE**. For example, even LLaMA-3.1-70B-Instruct, among the largest and most capable tested models, suffers from significant performance drops on AAVE queries. This pattern holds across the board, indicating that scaling model size alone is insufficient to address dialect-related performance disparities.

4 Discussions

This section investigates the potential reasons for AAVE's brittleness. We show that LLMs' brittleness with AAVE reasoning queries cannot be simply attributed to the lack of understanding of this dialect or simple lexical features. The nuanced con-

Models	SE	AAVE
LLaMA-3.1-70B-Instruct	9.4	17.5
Phi-3-Medium-128K-Instruct	5.9	12.5
Phi-3-Mini-128K-Instruct	7.1	15.9

Table 4: Averaged perplexities across instances calculated by different models on SE/AAVE ReDial.

versational norms of AAVE also play an important role in the problems for LLMs.

4.1 General Understanding and Morphosyntactic Features

One possible explanation for the performance drop is that LLMs cannot process AAVE dialect. We thus computed perplexities on ReDial AAVE vs. SAE prompts. Indeed, Table 4 confirms that LLMs exhibit higher perplexities on dialect than SE.

However, is the insufficient understanding the only reason that is causing LLMs' performance to drop? To answer it, we gradually inject typos in SE ReDial by replacing/deleting/adding words/characters, such that we make the input texts more difficult for LLMs (i.e., the measured perplexity goes up). We find that while these perturbations degrade model performance, the drop does not reach the severity observed with natural AAVE data on large-scale models (see full results in Section A.8). This discrepancy suggests that AAVE brittleness is not solely due to the general difficulty in processing. There are some dialect-specific reasons for the performance drop.

If language-agnostic processing ability cannot explain LLMs' brittleness, can we attribute the problem to morphosyntactic AAVE features? Following Ziems et al. (2022, 2023), we use morphosyntactic transformation rules to inject AAVE features into SE ReDial. We find that performance degradation generally intensifies as the density of AAVE-specific features increases when we gradually inject AAVE features into standard ReDial (see full results in Section A.9). This suggests that these features play a significant role in diminishing model performance.

However, even under the most extreme synthetic perturbations, performance drops are notably less severe than those observed with human-rewritten prompts. This underscores **the critical importance of our high-quality human-annotated dialect data ReDial** for evaluating LLM fairness and robustness. Synthetic rule-based transformations provide valuable insights, yet fail to capture the contextual depth of real-world dialect usage.

4.2 AAVE Conversational Norms

We use the mutual information between the token distributions of SE and AAVE ReDial to find that the top 5 most informative AAVE features in terms of distinguishing them from SE are *'*, *up*, *in*, *gon*, and *gotta*. We note that many of these features are not well-known AAVE-specific features (e.g., *up*). Through a further investigation of our dataset, we find that these lexicons are associated with phrase-level AAVE constructions. For instance, instead of saying *...encode the answer..* in Standard English, AAVE instruction says *...wrap it up...*. This finding is particularly interesting as we find that, in addition to previous linguistic observations of AAVE-specific morphosyntactic features, there are conversational norms of the dialect such as nuanced uses of phrases.

We compute Spearman's correlation between the frequency of the features we find in each instance and their corresponding performance drop. Indeed, these features play a significant role in predicting GPT-4o's performance degradation (r=-0.318, p<0.001). We further implement and analyze 12 rule-based AAVE features following Ziems et al. (2022) (details in Appendix A.11), which are well documented in linguistic literature such as "finna" as a maker of immediate future (Nguyen and Grieve, 2020). We notice that the influential lexical features are a subset of the feature set discovered by mutual information (i.e., some of the actual influential features are not encoded in synthetic transformation rules). Consequently, the influence of synthetic features is not as strong as those discovered by mutual information (r=-0.256, p<0.001). This means that **simple rule-based transformations that implement the most salient morphosyntactic AAVE features may not be able to capture rich, context-dependent use of the dialect and, therefore fall short in predicting LLMs' performance in real workflows.**

Aided by GPT-o1 preview to filter the vast amount of data, we conducted a linguistically informed analysis of LLMs' most frequent errors on AAVE. For the algorithmic tasks, grammatical constructions and non-standard verb forms (e.g., *finna'*, *'em*), omission of articles and auxiliary verbs may cause the model to misinterpret references and function naming conventions. For example, GPT-4o interprets *you gon' write a python function, python_function* as a general statement rather than a directive to name the function. On logic tasks, the frequent use of double negatives, zero copula, and inverted conditionals introduces structural ambiguities. For example, the construct *He don't take no breaks* can invert intended meanings, leading the model to misunderstand conditional statements. On math, informal expressions, unclear quantity references, and non-standard comparative cause erroneous parsing of numerical information and confusion over collective versus individual quantities. Informal phrasings like *half as much as he be runnin'* and ambiguous comparative expressions (*4 fewer boxes of apple pie than on Sunday*) can cause the model to misinterpret numerical relationships, resulting in errors when calculating totals, differences, or fractions. On the integrated task, phonetic spellings, colloquial connectors, and inverted word orders limit the model's understanding of concurrency, and stepwise instructions. **Such dialectal nuances highlight the necessity of our dataset and also call for more efforts to collect more human data for relevant purposes.**

5 Related Work

Dialect studies in natural language processing. Previous works on AAVE studies in natural language processing mostly focus on non-reasoning-heavy tasks such as POS tagging (Jørgensen et al., 2015, 2016), language identification and dependency parsing (Blodgett et al., 2016), automatic captioning (Tatman, 2017), and general language generation (Deas et al., 2023). AAVE is also found to be more likely to trigger false positives in hate speech identifiers (Davidson et al., 2019; Sap et al., 2019) due to specific word choices (Harris et al., 2022), be considered negative by automatic sentiment classifier (Groenwold et al., 2020), and cause covert biases in essential areas of social justice (Hofmann et al., 2024). Relevant studies (Ziems et al., 2022; Gupta et al., 2024) also find that rule-based AAVE feature perturbations can downgrade language model performance in a wide range of tasks covered by GLUE (Wang, 2018).

More generally, dialects in world languages pose challenges to natural language processing systems. Ziems et al. (2023) find that auto-encoder models are brittle on rule-based English dialect feature perturbations. Fleisig et al. (2024) report that English dialect speakers perceive responses generated by chatbots to be more negative than Standardized English (SE) prompts. Faisal et al. (2024) find that world dialects cause problems in tasks including dependency parsing (Scherrer et al., 2019) and machine translation (Mirzakhalov, 2021) on mBERT and XLM-R (Conneau et al., 2020).

Fairness and Robustness of Large Language Models. LLMs are widely testified to be both unfair and brittle. They introduce unfair performance (Huang et al., 2023; Dong et al., 2024) and cost (Petrov et al., 2024) to users across different languages, exacerbate social imbalance by marginalizing minority groups in various aspects including gender (Kotek et al., 2023; Fraser and Kiritchenko, 2024), race (Hofmann et al., 2024; Wang et al., 2024), and culture (Naous et al., 2023; Tao et al., 2024). They also show different performance to reasoning tasks in different languages (Huang et al. 2023, 2024; Ranaldi et al. 2024; inter alia). For the first time, our study provides extensive empirical evidence that LLMs exhibit unfairness in reasoning tasks. This bias specifically affects speakers of certain dialects within a single language.

Previous works report that LLMs are very brittle to slight variations of prompts by introducing typos or paraphrasing in SE (Elazar et al., 2021; Liang et al., 2022; Raj et al., 2022; Zhu et al., 2023b; Lin et al., 2024). In this work, we consider a novel application of using human-written perturbations in AAVE by asking humans to rewrite instances to their dialect and evaluate LLM robustness towards these natural perturbations, which have proven to cause LLMs to be more brittle than synthetic typo-style (Section 4.1) or linguistic-rule-based (Appendix A.9) perturbations.

6 Conclusion

Our study is the first attempt to objectively evaluate the dialect robustness and fairness of LLMs across reasoning tasks. We introduce **ReDial**, a dataset comprising over 1.2K parallel prompts in Standardized English and African American Vernacular English (AAVE) tailored to reasoning tasks: algorithm, logic, math, and integrated reasoning. Extensive empirical evidence on ReDial demonstrates that LLMs exhibit significant unfairness and brittleness when reasoning tasks are expressed in AAVE. These findings underscore the unfairness to dialect users and LLMs' brittleness with natural prompt variations with the same semantics. We advocate for further research to enhance dialect fairness and robustness of LLMs, ensuring equal service for all linguistic groups and demographics.

7 Limitations

First, as the first systematic framework for analyzing LLM bias in dialectal queries for reasoning tasks, we selected AAVE due to its linguistic significance and cultural impact. However, we recognize the vast diversity of dialects worldwide. The insights derived from AAVE may not generalize to other dialects. To ensure annotation quality and maintain the focus of our study, we concentrated on AAVE with high-quality human annotations. Future research could expand on our framework to encompass a wider range of dialects, generating more broadly applicable conclusions.

Second, our benchmark, ReDial, evaluates LLM performance across four categories of reasoning tasks using queries sampled from seven popular and well-documented benchmarks. While these tasks are representative of common reasoning challenges, we acknowledge that reasoning is a multifaceted domain with many additional categories and tasks that fall outside the scope of this study.

Third, we evaluated five representative LLM families in this study, including widely used and sota models. However, given the rapid proliferation of new LLMs, testing every model is infeasible. We hope that future research will use the ReDial benchmark to investigate fairness and reasoning robustness across a broader range of LLMs as they emerge.

Last but not least, while we present extensive empirical evidence demonstrating the performance drop of LLMs on dialectal queries, our study does not deeply investigate the underlying causes of these performance discrepancies or propose systematic methods to mitigate this bias. These topics exceed the scope of our work but are critical for addressing the inequities we have identified. Despite this limitation, we believe that ReDial provides a robust and systematic tool to help researchers explore these issues. The absence of immediate solutions should not detract from the significance of our findings, which lay the groundwork for future efforts to address fairness and robustness in LLMs.

8 Ethic Statement

ReDial is a collection of high-quality human-annotated translations: obtaining such data requires making clear design choices and poses ethical questions that we hereby address.

For data collection, we deliberately do not set hard constraints for annotator identity and demographic verification, recognizing there are no definite boundaries to identify dialects and their speakers (King, 2020). (King, 2020) further elaborate that the term "AAVE" itself is contested, with alternatives that could be used instead; in employing the term "AAVE", we adhere to the widely used terminology in related works on dialects and NLP (Ziems et al., 2022; Gupta et al., 2024). We corroborate the data quality by asking self-identified dialect speakers to cross-validate each others' answers.

We do not collect annotators' personal information; while we firmly commit to this rule to protect annotators' privacy, it makes it difficult to draw conclusions about how annotators' backgrounds shape their writing/individual-level variations. Further on the ethical aspect of data collection, we work with a data vendor that makes sure the recruitment and annotation adhere to high standards for and from the annotators. However, although we have a legal contract and we try our best to convey our guidelines and requirements, we admit that we do not have full control over how the vendor recruits people and conducts data annotation.

We also stress that the LLM validation stage in our quality control process is not completely trustworthy as even they are prone to hallucinations (Ji et al., 2023) and biases against minority groups (Xu et al., 2021; Fleisig et al., 2024; Smith et al., 2024; Wang et al., 2024). To mitigate this issue, we conduct full manual checks of every instance identified as invalid by an LLM so that no instance is rejected purely because of LLM decisions.

References

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. arXiv preprint arXiv:2404.14219.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. arXiv preprint arXiv:2303.08774.

Carolyn Temple Adger, Walt Wolfram, and Donna Christian. 2014. Dialects in schools and communities. Routledge.

Anthropic. The claude 3 model family: Opus, sonnet, haiku.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, et al. 2021. Program synthesis with large language models. arXiv preprint arXiv:2108.07732.

John Baugh. 2005. Linguistic profiling. In Black linguistics, pages 167–180. Routledge.

Su Lin Blodgett, Lisa Green, and Brendan O'Connor. 2016. Demographic dialectal variation in social media: A case study of African-American English. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1119–1130, Austin, Texas. Association for Computational Linguistics.

Jack K Chambers and Peter Trudgill. 1998. Dialectology. Cambridge University Press.

Guiming Hardy Chen, Shunian Chen, Ziche Liu, Feng Jiang, and Benyou Wang. 2024. Humans or llms as the judge? a study on judgement biases. arXiv preprint arXiv:2402.10669.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. arXiv preprint arXiv:2107.03374.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 8440–8451, Online. Association for Computational Linguistics.

Jay Cunningham, Su Lin Blodgett, Michael Madaio, Hal Daumé Iii, Christina Harrington, and Hanna Wallach. 2024. Understanding the impacts of language technologies' performance disparities on african american language speakers. In Findings of the Association for Computational Linguistics ACL 2024, pages 12826–12833.

Thomas Davidson, Debasmita Bhattacharya, and Ingmar Weber. 2019. Racial bias in hate speech and abusive language detection datasets. arXiv preprint arXiv:1905.12516.

Nicholas Deas, Jessica Grieser, Shana Kleiner, Desmond Patton, Elsbeth Turcan, and Kathleen McKeown. 2023. Evaluation of African American language bias in natural language generation. In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pages 6805–6824, Singapore. Association for Computational Linguistics.

Guoliang Dong, Haoyu Wang, Jun Sun, and Xinyu Wang. 2024. Evaluating and mitigating linguistic discrimination in large language models. arXiv preprint arXiv:2404.18534.

Rotem Dror, Gili Baumer, Segev Shlomov, and Roi Reichart. 2018. The hitchhiker's guide to testing statistical significance in natural language processing. In Proceedings of the 56th annual meeting of the association for computational linguistics (volume 1: Long papers), pages 1383–1392.

Anna Drożdżowicz and Yael Peled. 2024. The complexities of linguistic discrimination. Philosophical Psychology, pages 1–24.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. arXiv preprint arXiv:2407.21783.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and improving consistency in pretrained language models. Transactions of the Association for Computational Linguistics, 9:1012–1031.

Fahim Faisal, Orevaoghene Ahia, Aarohi Srivastava, Kabir Ahuja, David Chiang, Yulia Tsvetkov, and Antonios Anastasopoulos. 2024. Dialectbench: A nlp benchmark for dialects, varieties, and closely-related languages. arXiv preprint arXiv:2403.11009.

Eve Fleisig, Genevieve Smith, Madeline Bossi, Ishita Rustagi, Xavier Yin, and Dan Klein. 2024. Linguistic bias in chatgpt: Language models reinforce dialect discrimination. arXiv preprint arXiv:2406.08818.

Kathleen C Fraser and Svetlana Kiritchenko. 2024. Examining gender and racial bias in large vision-language models using a novel dataset of parallel images. arXiv preprint arXiv:2402.05779.

Víctor González-Calatayud, Paz Prendes-Espinosa, and Rosabel Roig-Vila. 2021. Artificial intelligence for student assessment: A systematic review. Applied sciences, 11(12):5467.

Sophie Groenwold, Lily Ou, Aesha Parekh, Samhita Honnavalli, Sharon Levy, Diba Mirza, and William Yang Wang. 2020. Investigating African-American Vernacular English in transformer-based text generation. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 5877–5883, Online. Association for Computational Linguistics.

Jeffrey Grogger. 2011. Speech patterns and racial wage inequality. Journal of Human resources, 46(1):1–25.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. 2023. Textbooks are all you need. arXiv preprint arXiv:2306.11644.

Abhay Gupta, Philip Meng, Ece Yurtseven, Sean O'Brien, and Kevin Zhu. 2024. Aavenue: Detecting llm biases on nlu tasks in aave via a novel benchmark. arXiv preprint arXiv:2408.14845.

Simeng Han, Hailey Schoelkopf, Yilun Zhao, Zhenting Qi, Martin Riddell, Luke Benson, Lucy Sun, Ekaterina Zubova, Yujie Qiao, Matthew Burtell, et al. 2022. Folio: Natural language reasoning with first-order logic. arXiv preprint arXiv:2209.00840.

Camille Harris, Matan Halevy, Ayanna Howard, Amy Bruckman, and Diyi Yang. 2022. Exploring the role of grammar and word choice in bias toward African American English (AAE) in hate speech classification. In Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency, pages 789–798.

Valentin Hofmann, Pratyusha Ria Kalluri, Dan Jurafsky, and Sharese King. 2024. Dialect prejudice predicts ai decisions about people's character, employability, and criminality. arXiv preprint arXiv:2403.00742.

Sture Holm. 1979. A simple sequentially rejective multiple test procedure. Scandinavian journal of statistics, pages 65–70.

Haoyang Huang, Tianyi Tang, Dongdong Zhang, Wayne Xin Zhao, Ting Song, Yan Xia, and Furu Wei. 2023. Not all languages are created equal in llms: Improving multilingual capability by cross-lingual-thought prompting. arXiv preprint arXiv:2305.07004.

Jie Huang and Kevin Chen-Chuan Chang. 2022. Towards reasoning in large language models: A survey. arXiv preprint arXiv:2212.10403.

Zixian Huang, Wenhao Zhu, Gong Cheng, Lei Li, and Fei Yuan. 2024. Mindmerger: Efficient boosting llm reasoning in non-english languages. arXiv preprint arXiv:2405.17386.

M Huth. 2004. Logic in Computer Science: Modelling and reasoning about systems. Cambridge University Press.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. ACM Computing Surveys, 55(12):1–38.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. arXiv preprint arXiv:2310.06825.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. arXiv preprint arXiv:2401.04088.

Xiaomeng Jin, Bhanukiran Vinzamuri, Sriram Venkatapathy, Heng Ji, and Pradeep Natarajan. 2023. Adversarial robustness for large language NER models using disentanglement and word attributions. In Findings of the Association for Computational Linguistics: EMNLP 2023, pages 12437–12450, Singapore. Association for Computational Linguistics.

Anna Jørgensen, Dirk Hovy, and Anders Søgaard. 2015. Challenges of studying and processing dialects in social media. In Proceedings of the Workshop on Noisy User-generated Text, pages 9–18, Beijing, China. Association for Computational Linguistics.

Anna Jørgensen, Dirk Hovy, and Anders Søgaard. 2016. Learning a POS tagger for AAVE-like language. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1115–1120, San Diego, California. Association for Computational Linguistics.

Anjali Kantharuban, Jeremiah Milbauer, Emma Strubell, and Graham Neubig. 2024. Stereotype or personalization? user identity biases chatbot recommendations. arXiv preprint arXiv:2410.05613.

Sharese King. 2020. From african american vernacular english to african american language: Rethinking the study of race and language in african americans' speech. Annual Review of Linguistics, 6(1):285–300.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. Advances in neural information processing systems, 35:22199–22213.

Hadas Kotek, Rikker Dockum, and David Sun. 2023. Gender bias and stereotypes in large language models. In Proceedings of the ACM collective intelligence conference, pages 12–24.

Emanuele La Malfa, Aleksandar Petrov, Simon Frieder, Christoph Weinhuber, Ryan Burnell, Raza Nazar, Anthony G Cohn, Nigel Shadbolt, and Michael Wooldridge. 2024. Language-models-as-a-service: Overview of a new paradigm and its challenges. Journal of Artificial Intelligence Research, 80:1497–1523.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Tianhao Wu, Banghua Zhu, Joseph E Gonzalez, and Ion Stoica. 2024. From crowdsourced data to high-quality benchmarks: Arena-hard and benchbuilder pipeline. arXiv preprint arXiv:2406.11939.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. 2022. Holistic evaluation of language models. arXiv preprint arXiv:2211.09110.

Fangru Lin, Emanuele La Malfa, Valentin Hofmann, Elle Michelle Yang, Anthony Cohn, and Janet B Pierrehumbert. 2024. Graph-enhanced large language models in asynchronous plan reasoning. arXiv preprint arXiv:2402.02805.

Rosina Lippi-Green. 1997. What we talk about when we talk about ebonics: Why definitions matter. The Black Scholar, 27(2):7–11.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2023. Is your code generated by chatGPT really correct? rigorous evaluation of large language models for code generation. In Thirty-seventh Conference on Neural Information Processing Systems.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, et al. 2024. Self-refine: Iterative refinement with self-feedback. Advances in Neural Information Processing Systems, 36.

Emanuele La Malfa and Marta Kwiatkowska. 2022. The king is naked: On the notion of robustness for natural language processing. Proceedings of the AAAI Conference on Artificial Intelligence, 36(10):11047–11057.

Joshua L Martin and Kelly Elizabeth Wright. 2022. Bias in automatic speech recognition: The case of african american language. Applied Linguistics, 44(4):613–630.

Douglas S Massey and Garvey Lundy. 2001. Use of black english and racial discrimination in urban housing markets: New methods and findings. Urban affairs review, 36(4):452–469.

Quinn McNemar. 1947. Note on the sampling error of the difference between correlated proportions or percentages. Psychometrika, 12(2):153–157.

Dan Milmo. 2023. Chatgpt passes 100 million users, making it the fastest-growing app in history. The Guardian. Accessed: 2024-09-27.

Jamshidbek Mirzakhalov. 2021. Turkic interlingua: a case study of machine translation in low-resource languages. Master's thesis, University of South Florida.

Milad Moradi and Matthias Samwald. 2021. Evaluating the robustness of neural language models to input perturbations. CoRR, abs/2108.12237.

Tarek Naous, Michael J Ryan, Alan Ritter, and Wei Xu. 2023. Having beer after prayer? measuring cultural bias in large language models. arXiv preprint arXiv:2305.14456.

Dong Nguyen and Jack Grieve. 2020. Do word embeddings capture spelling variation? In Proceedings of the 28th International Conference on Computational Linguistics, pages 870–881, Barcelona, Spain (Online). International Committee on Computational Linguistics.

OpenAI. Openai o1 system card.

OpenAI. 2024. Gpt-4o system card. ArXiv.

Mihir Parmar, Nisarg Patel, Neeraj Varshney, Mutsumi Nakamura, Man Luo, Santosh Mashetty, Arindam Mitra, and Chitta Baral. 2024. Logicbench: Towards systematic evaluation of logical reasoning ability of large language models. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 13679–13707.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems? arXiv preprint arXiv:2103.07191.

Aleksandar Petrov, Emanuele La Malfa, Philip Torr, and Adel Bibi. 2024. Language model tokenizers introduce unfairness between languages. Advances in Neural Information Processing Systems, 36.

Thomas Purnell, William Idsardi, and John Baugh. 1999. Perceptual and phonetic experiments on american english dialect identification. Journal of language and social psychology, 18(1):10–30.

Shuofei Qiao, Yixin Ou, Ningyu Zhang, Xiang Chen, Yunzhi Yao, Shumin Deng, Chuanqi Tan, Fei Huang, and Huajun Chen. 2022. Reasoning with language model prompting: A survey. arXiv preprint arXiv:2212.09597.

Harsh Raj, Domenic Rosati, and Subhabrata Majumdar. 2022. Measuring reliability of large language models through semantic consistency. arXiv preprint arXiv:2211.05853.

Leonardo Ranaldi, Giulia Pucci, Barry Haddow, and Alexandra Birch. 2024. Empowering multi-step reasoning across languages via program-aided language models. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 12171–12187.

John R Rickford and Sharese King. 2016. Language and linguistics on trial: Hearing rachel jeantel (and other vernacular speakers) in the courtroom and beyond. Language, pages 948–988.

Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The risk of racial bias in hate speech detection. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 1668–1678, Florence, Italy. Association for Computational Linguistics.

Yves Scherrer, Tanja Samardžić, and Elvira Glaser. 2019. Digitising swiss german: how to process and study a polycentric spoken language. Language Resources and Evaluation, 53(4):735–769.

Lin Shi, Weicheng Ma, and Soroush Vosoughi. 2024. Judging the judges: A systematic investigation of position bias in pairwise comparative assessments by llms. arXiv preprint arXiv:2406.07791.

Genevieve Smith, Eve Fleisig, Madeline Bossi, Ishita Rustagi, and Xavier Yin. 2024. Standard language ideology in ai-generated language. arXiv preprint arXiv:2406.08726.

Yan Tao, Olga Viberg, Ryan S Baker, and René F Kizilcec. 2024. Cultural bias and cultural alignment of large language models. PNAS Nexus, 3(9):pgae346.

Rachael Tatman. 2017. Gender and dialect bias in YouTube's automatic captions. In Proceedings of the First ACL Workshop on Ethics in Natural Language Processing, pages 53–59, Valencia, Spain. Association for Computational Linguistics.

Alex Wang. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. arXiv preprint arXiv:1804.07461.

Angelina Wang, Jamie Morgenstern, and John P Dickerson. 2024. Large language models cannot replace human participants because they cannot portray identity groups. arXiv preprint arXiv:2402.01908.

PC Wason. 1972. Psychology of Reasoning: Structure and Content. Cambridge/Harvard University Press.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems, 35:24824–24837.

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. 2023. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. arXiv preprint arXiv:2307.02477.

Albert Xu, Eshaan Pathak, Eric Wallace, Suchin Gururangan, Maarten Sap, and Dan Klein. 2021. Detoxifying language models risks marginalizing minority voices. arXiv preprint arXiv:2104.06390.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2024. Tree of thoughts: Deliberate problem solving with large language models. Advances in Neural Information Processing Systems, 36.

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

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

Kaijie Zhu, Jiaao Chen, Jindong Wang, Neil Zhenqiang Gong, Diyi Yang, and Xing Xie. 2023a. Dyval: Dynamic evaluation of large language models for reasoning tasks. In The Twelfth International Conference on Learning Representations.

Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Yue Zhang, Neil Zhenqiang Gong, et al. 2023b. Promptbench: Towards evaluating the robustness of large language models on adversarial prompts. arXiv preprint arXiv:2306.04528.

Caleb Ziems, Jiaao Chen, Camille Harris, Jessica Anderson, and Diyi Yang. 2022. Value: Understanding dialect disparity in nlu. arXiv preprint arXiv:2204.03031.

Caleb Ziems, William Held, Jingfeng Yang, Jwala Dhamala, Rahul Gupta, and Diyi Yang. 2023. Multi-VALUE: A framework for cross-dialectal English NLP. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 744–768, Toronto, Canada. Association for Computational Linguistics.

A Appendix

A.1 Source Dataset Illustration

A.1.1 Algorithm

Original HumanEval

```
from typing import List


def has_close_elements(numbers:
List[float], threshold: float)
-> bool:
    """ Check if in given list of
    numbers, are any two numbers
    closer to each other than
    given threshold.
    >>> has_close_elements([1.0,
    2.0, 3.0], 0.5)
    False
    >>> has_close_elements([1.0,
    2.8, 3.0, 4.0, 5.0, 2.0], 0.3)
    True
    """
```

InstructHumanEval Used in the Paper

Write a function has_close_elements(numbers: List[float], threshold: float) -> bool to solve the following problem:
Check if in given list of numbers, are any two numbers closer to each other than given threshold.
>>> has_close_elements([1.0, 2.0, 3.0], 0.5)
False
>>> has_close_elements([1.0, 2.8, 3.0, 4.0, 5.0, 2.0], 0.3)
True

MBPP

Write a python function to remove first and last occurrence of a given character from the string.
Your code should pass these tests:
assert remove_Occ("hello","l") == "heo"
assert remove_Occ("abcda","a") == "bcd"
assert remove_Occ("PHP","P") == "H"

A.1.2 Logic

LogicBench

If an individual consumes a significant amount of water, they will experience a state of hydration. Conversely, if excessive amounts of sugar are ingested, a sugar crash will ensue. It is known that at least one of the following statements is true: either the Jane consumes ample water or she will not experience a sugar crash. However, the actual veracity of either statement remains ambiguous, as it could be the case that only the first statement is true, only the second statement is true, or both statements are true.
Can we say at least one of the following must always be true? (a) she will feel hydrated and (b) she doesn't eat too much sugar

Folio

Consider the following premises: "People in this club who perform in school talent shows often attend and are very engaged with school events. People in this club either perform in school talent shows often or are inactive and disinterested community members. People in this club who chaperone high school dances are not students who attend the school. All people in this club who are inactive and disinterested members of their community chaperone high school dances. All young children and teenagers in this club who wish to further their academic careers and educational opportunities are students who attend the school. Bonnie is in this club and she either both attends and is very engaged with school events and is a student who attends the school or is not someone who both attends and is very engaged with school events and is not a student who attends the school."
Assuming no other commonsense or world knowledge, is the sentence "Bonnie performs in school talent shows often." necessarily true, necessarily false, or neither? Answer either "necessarily true", "necessarily false", or "neither".

A.1.3 Math

GSM8K

Given a mathematics problem, determine the answer. Simplify your answer as much as possible and encode the final answer in <answer></answer> (e.g., <answer>1</answer>).
Question: Janet's ducks lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder at the farmers' market daily for $2 per fresh duck egg. How much in dollars does she make every day at the farmers' market?
Answer:

SVAMP

Given a mathematics problem, determine the answer. Simplify your answer as much as possible and encode the final answer in <answer></answer> (e.g., <answer>1</answer>).
Question: Winter is almost here and most animals are migrating to warmer countries. There are 41 bird families living near the mountain. If 35 bird families flew away to asia and 62 bird families flew away to africa How many more bird families flew away to africa than those that flew away to asia?
Answer:

A.1.4 Comprehensive

AsyncHow

To create a video game, here are the steps and the times needed for each step.
Step 1. Learn the basics of programming (180 days)
Step 2. Learn to use a language that is used in games (60 days)
Step 3. Learn to use an existing game engine (30 days)
Step 4. Program the game (90 days)
Step 5. Test the game (30 days)

These ordering constraints need to be obeyed when executing above steps:
Before starting step 2, complete step 1.
Before starting step 3, complete step 1.
Before starting step 4, complete step 2.
Before starting step 4, complete step 3.
Before starting step 5, complete step 4.

Question: Assume that you need to execute all the steps to complete the task and that infinite resources are available. What is the shortest possible time to create a video game? Answer the time in double quotes.
Answer:

A.2 ReDial Samples

Algorithm

Standardized

Write a function python_function(numbers: List[float], threshold: float) − > bool to realize the following functionality:
Check if in given list of numbers, are any two numbers closer to each other than given threshold.
>>> python_function([1.0, 2.0, 3.0], 0.5)
False
>>> python_function([1.0, 2.8, 3.0, 4.0, 5.0, 2.0], 0.3)
True
Generate a Python function to solve this problem. Ensure the generated function is named as python_function.

Algorithm

AAVE

Aight, so here you gonna write a function called python_function(numbers: List[float], threshold: float) $->$ bool that gon' do this following functionality:
Aight, Listen. Say you got a list of numbers yeah? Now, we trynna see if any two of 'em numbers is closer to each other than a number you give, feel me?So, this is what we 'bout to do:
$>>>$ python_function([1.0, 2.0, 3.0], 0.5)
False
That's gon' give you False cuz ain't none of 'em numbers close enough.But, if you hit it like:
$>>>$ python_function([1.0, 2.8, 3.0, 4.0, 5.0, 2.0], 0.3)
True
Bet you gettin' True, cuz this time some of 'em numbers real tight.
You gotta whip up a Python function to handle this problem. You gon' make sure the function name right, which gotta python_function.

Math

Standardized

Given a mathematics problem, determine the answer. Simplify your answer as much as possible and encode the final answer in $< answer >< /answer >$ (e.g., $< answer > 1 < /answer >$).
Question: John is raising money for a school trip. He has applied for help from the school, which has decided to cover half the cost of the trip. How much money is John missing if he has $50 and the trip costs $300?
Answer:

Math

AAVE

Bet, so here's whatsup. Youn finna get a math problem, and you gon' tryna find the answer out. You gotta simplify that answer as much as possible tehn wrap it up inside $< answer >< /answer >$ (somethin' like this:, $< answer > 1 < /answer >$).
Question: John been raisin' money fo' a school trip. He done ask the school fo' help, and they decided they gon' be coverin' half the trip cost. How much money John be missin' if he got $50, and the trip cost $300.
Answer:

Logic

Standardized

Consider the following premises: "All bears in zoos are not wild.
Some bears are in zoos. "
Assuming no other commonsense or world knowledge, is the sentence "Not all bears are wild." necessarily true, necessarily false, or neither? Answer either "necessarily true", "necessarily false", or "neither". Encode the final answer in $< answer >< /answer >$ (e.g., $< answer >$necessarily true$< /answer >$).

Logic

AAVE

Aight, check this. You got 'em premises right here: "All bears in zoos ain't considered wild.
There are some bears livin' in zoos. "
Ain't no using no other commonsense or world knowledge, you gon' try find out if the sentence "Not every bear out there be wild." necessarily true, necessarily false, or neither? Pick either "necessarily true", "necessarily false", or "neither". Then wrap that answer up in $< answer >< /answer >$ (e.g., $< answer >$necessarily true$< /answer >$).

Comprehensive

Standardized

To try fishing for the first time, here are the steps and the times needed for each step
Step 1. drive to the outdoor store (10 minutes)
Step 2.compare fishing poles (30 minutes)
Step 3. buy a fishing pole (5 minutes)
Step 4. buy some bait (5 minutes)
Step 5. drive to a lake (20 minutes)
Step 6. rent a small boat (15 minutes)

These ordering constraints need to be obeyed when executing above steps:
Step 1 must precede step 2.
Step 2 must precede step 3.
Step 2 must precede step 4.
Step 3 must precede step 5.
Step 4 must precede step 5
Step 5 must precede step 6.

Question: Assume that you need to execute all the steps to complete the task and that infinite resources are available. What is the shortest possible time to complete this task? What is the shortest possible time to complete this task? Encode the final answer in $< answer >< /answer >$ (e.g., $< answer >$1 min$< /answer >$).
Answer:

Comprehensive

AAVE

If you finna go fish for the first time, here's what you got to know and the times you need for each step.
Step 1. To kick things off, pull up to the outdoor store (10 minutes)
Step 2. Check out which one of them fishing poles is good and which one is not (30 minutes)
Step 3. Cop a fishing pole (5 minutes)
Step 4.Get yourself some bait as well (5 minutes)
Step 5. Head out to a lake (20 minutes)
Step 6.rent yourself a small boat (15 minutes)

These ordering constraints gotta be followed when you doin' 'em steps above: You gotta deal with 1 before hittin' the 2.
You gotta deal with 2 before hittin' the 3.
You gotta deal with 2 before hittin' the 4.
You gotta deal with 3 before hittin' the 5.
You gotta deal with 4 before hittin' the 5.
You gotta deal with 5 before hittin' the 6.

Question: Assumin' you outta do all 'em steps to finish up the task, and you got infinite resources. What the shortest time be to knock this task out? Wrap that answer up in $< answer >< /answer >$ (e.g., $< answer >$1 min$< /answer >$).
Answer:

A.3 Rubrics

A.3.1 Employment Information

We work with data vendors to employ 13 annotators in total for our task. For algorithm instance annotation, we specifically hire annotators with computer science backgrounds. Annotators are self-identified as proficient speakers of African American Vernacular English. We do not pose any hard constraints in verifying dialect identity as previous studies do (e.g., (Ziems et al., 2023)). We note even within a dialect there can be significant variations on the individual level and that we want to avoid homogenization and over-simplification of the dialect (King, 2020). Instead, we ask self-identified annotators to cross-check each other's annotations and modify if they sound unnatural.

Details of employment are shown below.

Information Collected We do not collect personally identifiable information from our annotators (e.g., name, age, etc). We only collect the annotators' responses to our consent form and their annotations of our data.

Risk and Consent We note that our base datasets are from publicly available, widely used, peer-reviewed datasets that adhere to peer-review regulations. Moreover, our tasks are mainly centered around reasoning, which does not concern sensitive information per se. In addition, we make sure that annotators understand the risks of the annotation (i.e., although we have tried our best to ensure the safety of the data, it is still possible that they may feel uncomfortable in the annotation) and their right to exit the task during the process by signing a consent form prior to the start of the task.

Compensation We offer payment to annotators with hourly rates higher than the U.S. federal minimum wage.

No AI Assistant We explicitly inform our annotators that they should not reply on any AI assistant tools to help them complete the task. To further ensure this, we design our annotation platform to disallow copy and paste. The default annotation area for annotators is the original text, which means that it is easier for annotators to simply edit the text than querying AI assistants.

A.3.2 Annotation Guideline

You need to translate/rephrase/localize the task input in a way that is natural to the speakers of your dialect without changing the intention of the prompts. You should not change named entities, numbers, equations, variable names and other formal devices that are not natural language per se or those that would affect the intention of the prompts. The translation does not need to be grammatical or acceptable in standard English. Rather, it should accurately reflect the features of their dialects. You can add or delete some functional content to make the prompts sound more natural (e.g., adding fillers). However, you should keep the vital information complete and unchanged.

You should NOT change information that would invalidate the output given the question. If you are unsure about any specific parts, leave them unchanged. Especially, you should not change the following parts:

(i) numbers (e.g. 180 in 180 days)

(ii) units (e.g. days in 180 days)

(iii) equations and symbols (e.g., \[f(x) = \left \{ \begin{array}{cl} ax+3, & \text{ if }$x > 2$ in Let \[$f(x) =$ \left \{ \begin{array}{cl} $ax + 3$, & \text{ if }$x > 2$)

(iv) proper nouns (e.g., Natalia in Natalia sold clips to 48 of her friends)

(v) function names, variables, data types, and input-output examples (e.g., >>> has_close_elements($[1.0, 2.0, 3.0]$, 0.5) False >>> has_close_elements($[1.0, 2.8, 3.0, 4.0, 5.0, 2.0]$, 0.3) True in Check if in given list of numbers, are any two numbers closer to each other than given threshold. >>> has_close_elements($[1.0, 2.0, 3.0]$, 0.5) False >>> has_close_elements($[1.0, 2.8, 3.0, 4.0, 5.0, 2.0]$, 0.3) True)

A.4 Data Quality Verification

After we conduct human validations for *naturalness* and *correctness* of prompts, we conduct the final round sanity check with GPT-4o. We prompt GPT-4o with temperature 0.7 and sample three instances for each query. We manually inspect instances again where all of the answers suggest that they are invalid paraphrases of the original prompts.

User prompt

You will be given two prompts, one in Standard English and one in African American English. Determine whether the African American English prompt is a valid paraphrase of the Standard English prompt. Ignore the semantic validaty of the Standard English prompt.
Standard English: "[SAE_PROMPT]"
African American English: "[AAVE_PROMPT]"
Is the African American English prompt a valid paraphrase of the Standard English prompt?

A.5 Implementation Details

A.5.1 Dataset Implementation

For Algorithm, we unify the prompts by substituting all function names as python_function to avoid as much memorization as possible. We also manually corrected instances in HumanEval where the task descriptions were not precise enough (e.g., when the output data structure specified in the docstring is different from the one specified in the function heading). We also slightly modified some instructions in algorithm datasets without changing their intention to make sure our prompts are coherent (e.g., changing *to solve the following problem* to *to realize the following functionality*).

For other tasks, we unify the task output by asking LLMs to encode answers in $< answer ><$ $/answer >$ to enable easy parsing. All details can be found in ReDial dataset files.

A.5.2 Inference Implementation

We set temperature=0 and max new token as 4096 for all models at inference time unless specified in the main paper. We run experiments on GPT-4o/4/3.5 via Azure OpenAI service. We evaluate all other models via Azure Machine Learning Studio API for main results. Experiments run in the analysis part are hosted on 4 A100 with 80GB memory each.

A.6 Results for Non-zero Temperature

We vary the temperature by 0, 0.5, 0.7, and 1 on GPT-4o/4/3.5-turbo and Phi-3-Mini/Medium-128K-Instruct. When the temperature is not 0, we sample 3 answers per query and take average pass rates as results for corresponding settings. Results are in Figure 3.

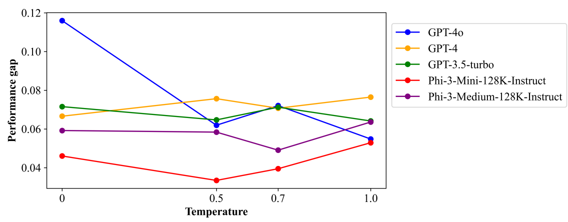

Figure 3: We vary the temperature by 0, 0.5, 0.7, 1 and report the performance gap between Standardized and AAVE ReDial.

We find that increasing temperature reduces the gap for GPT-4o in general, but does not affect other models' performance as much. Even when the performance gap is reduced, increasing temperature cannot cancel the gap.

A.7 Full Results on ReDial

we present the complete results on Redial. Specifically, Table 5 provides the detailed results for Algorithm, Table 6 covers the results for Logic, Table 7 reports the results for Math, and Table 8 reports the results for Integrated Tasks.

Model	Setting	HumanEval		MBPP	
		Original	AAVE	Original	AAVE
GPT-o1 🔒	Vanilla	0.860	$0.860_{(+)0.000}$	0.773	$0.787_{(+)0.013}$
GPT-4o 🔒	Vanilla	0.872	$0.811_{(-)0.061}$	0.700	$0.707_{(+)0.007}$
	CoT	0.841	$0.805_{(-)0.037}$	0.693	$0.713_{(+)0.02}$
GPT-4 🔒	Vanilla	0.780	$0.744_{(-)0.037}$	0.700	$0.700_{(-)-0.0}$
	CoT	0.750	$0.707_{(-)0.043}$	0.693	$0.500_{(-)0.193}$
GPT-3.5-turbo 🔒	Vanilla	0.640	$0.622_{(-)0.018}$	0.667	$0.640_{(-)0.027}$
	CoT	0.616	$0.591_{(-)0.024}$	0.680	$0.507_{(-)0.173}$
Claude-Sonnet 🔒	Vanilla	0.787	$0.848_{(+)0.061}$	0.753	$0.760_{(+)0.007}$
	CoT	0.793	$0.726_{(-)0.067}$	0.753	$0.747_{(-)0.007}$
LLaMA-3.1-70B-Instruct	Vanilla	0.744	$0.726_{(-)0.018}$	0.707	$0.573_{(-)0.133}$
	CoT	0.738	$0.689_{(-)0.049}$	0.707	$0.613_{(-)0.093}$
LLaMA-3-70B-Instruct	Vanilla	0.689	$0.671_{(-)0.018}$	0.673	$0.613_{(-)0.06}$
	CoT	0.720	$0.665_{(-)0.055}$	0.673	$0.627_{(-)0.047}$
LLaMA-3-8B-Instruct	Vanilla	0.530	$0.524_{(-)0.006}$	0.540	$0.493_{(-)0.047}$
	CoT	0.537	$0.512_{(-)0.024}$	0.527	$0.440_{(-)0.087}$
Mixtral-8x7B-Instruct-v0.1	Vanilla	0.402	$0.390_{(-)0.012}$	0.507	$0.413_{(-)0.093}$
	CoT	0.396	$0.396_{(-)-0.0}$	0.547	$0.427_{(-)0.12}$
Mistral-7B-Instruct-v0.3	Vanilla	0.268	$0.268_{(-)-0.0}$	0.400	$0.240_{(-)0.16}$
	CoT	0.262	$0.274_{(+)0.012}$	0.367	$0.213_{(-)0.153}$
Phi-3-Medium-128K-Instruct	Vanilla	0.530	$0.518_{(-)0.012}$	0.560	$0.340_{(-)0.22}$
	CoT	0.530	$0.573_{(+)0.043}$	0.567	$0.327_{(-)0.24}$
Phi-3-Small-128K-Instruct	Vanilla	0.598	$0.329_{(-)0.268}$	0.633	$0.167_{(-)0.467}$
	CoT	0.585	$0.293_{(-)0.293}$	0.553	$0.087_{(-)0.467}$
Phi-3-Mini-128K-Instruct	Vanilla	0.549	$0.482_{(-)0.067}$	0.567	$0.367_{(-)0.2}$
	CoT	0.567	$0.530_{(-)0.037}$	0.587	$0.347_{(-)0.24}$

Table 5: All results for **Algorithm**.

A.8 Perplexity of Typos vs. Natural AAVE Text

We gradually inject typos in SE ReDial by replacing/deleting/adding words/characters, such that we make the input texts more difficult for LLMs (i.e., the measured perplexity goes up). We find that while these perturbations degrade model performance, the drop did not reach the severity observed with natural AAVE data on large-scale models (see full results in Section A.8). This discrepancy suggests that AAVE brittleness is not solely due to the general difficulty in processing. Interestingly, larger models such as LLaMA-3.1-70B-Instruct and Phi-3-Medium-128K-Instruct degrade less on synthetic noise yet still struggle disproportionately with authentic AAVE data. This suggests their brittleness stems from reliance on SE priors and limited encoding of AAVE-specific discourse structures.

A.9 Multivalue Perturbation

Since the unfamiliarity of data cannot explain the whole picture, how much can we attribute the failure to AAVE-specific features? We use the rule-based transformation method in (Ziems et al., 2023) to inject AAVE features into our dataset for synthetic probing. We compare GPT-4o/4/3.5 and Phi-3-Medium/Mini-128k-Instruct performance in feature densities of $\{0, 0.25, 0.5, 0.75, 1\}$ and run the same setting as the main experiment.

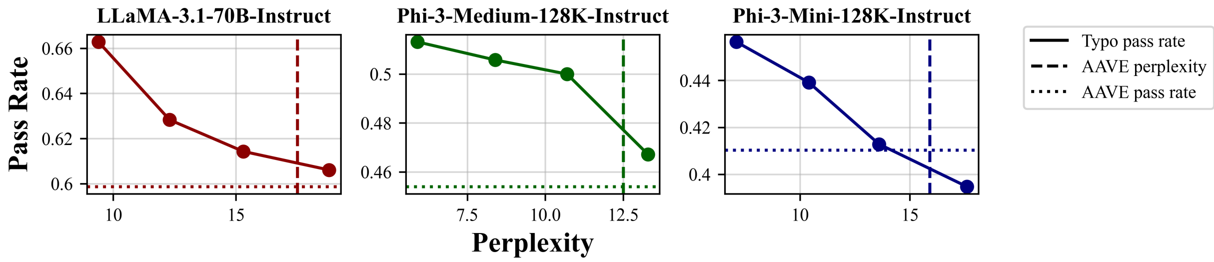

Figure 4: Model performance on misspelled SE compared to human-written AAVE data. We gradually add noise to SE ReDial to increase its perplexities until they surpass the perplexity of AAVE ReDial and report the models' performance on every perturbation level. Horizontal and vertical lines refer to model pass rates/perplexities on AAVE ReDial respectively. Larger LLMs (i.e., LLaMA-3.1-70B-Instruct and Phi-3-Medium-128K-Instruct) perform worse on AAVE than on perturbed text with a similar perplexity level.

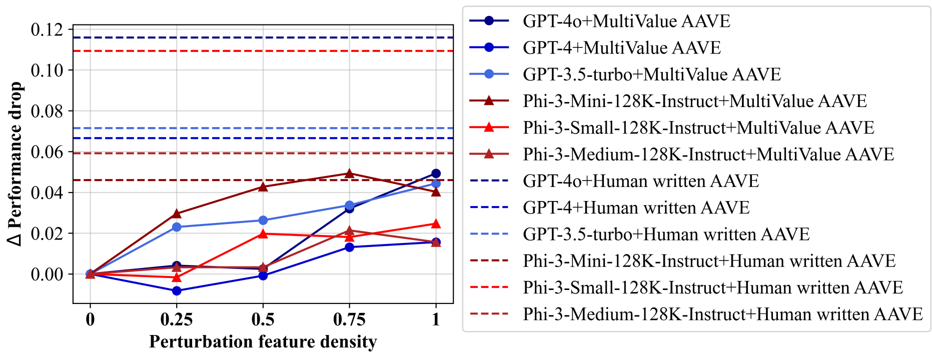

Figure 5: Perturbation with AAVE features. We control perturbation feature densities at $\{0, 0.25, 0.5, 0.75, 1\}$ to gradually inject AAVE features using rule-based transformations.

Results are shown in Figure 5. On the one hand, we find that models generally show increasing performance drops with increasing feature density, which means that AAVE-specific features do contribute to model performance drops. On the other hand, even drops caused by the strongest perturbation are generally far from the drops caused by human-rewritten prompts. This shows the limitation of previous methods in revealing LLM robustness based on synthetic data as there can be more influential factors than what lexico-syntactic rules can capture. Phi-3-Mini-128K-Instruct is again an outlier here, being that it is the only model that has a stronger performance drop in feature injections compared to human-written dialect data.

A.10 Qualitative Analysis of Standardized Prompting

We observe a sensible gap between model performance on Standardized ReDial with zero-shot prompting and AAVE ReDial with standardization prompting. In this section, we qualitatively compare GPT-4o's outputs in these two settings, the model with the best absolute overall performance, and examine its errors. We focus on the math subset of ReDial and identify three key error patterns: **wrong question rephrasing**, **distraction by irrelevant information**, and **failure to execute all steps**.

Wrong question rephrasing. GPT-4o wrongly phrases question '*Jame ... How many years have they got between them now if in 8 years his cousin will be 5 years younger than twice his age?*' to '*James ... How old is his cousin now?*', which changes the question of age gap to absolute age.

Distraction by irrelevant information. GPT-4o gets distracted by task-irrelevant information after AAVE standardization while the distraction is not observed in Standardized ReDial. For instance, in '*Say we got 8 different books and 10 different movies in the crazy silly school series. How many more movies than books is there gon be in the crazy silly school series if you read 19 books and watched 61 movies?*', books that have been read and movies that have been watched are not associated with the answer. Although GPT-4o can ignore irrelevant information in Standardized ReDial, it gets distracted after AAVE standardization, which shows the brittleness of its reasoning ability.

Failure to execute all the steps. GPT-4o some-

Model	Setting	Folio		LogicBench	
		Original	AAVE	Original	AAVE
GPT-o1 🔒	Vanilla	0.963	$0.938_{(-)0.025}$	0.810	$0.715_{(-)0.095}$
GPT-4o 🔒	Vanilla	0.938	$0.870_{(-)0.068}$	0.720	$0.685_{(-)0.035}$
	CoT	0.938	$0.926_{(-)0.012}$	0.715	$0.645_{(-)0.070}$
GPT-4 🔒	Vanilla	0.858	$0.796_{(-)0.062}$	0.745	$0.710_{(-)0.035}$
	CoT	0.864	$0.759_{(-)0.105}$	0.735	$0.730_{(-)0.005}$
GPT-3.5-turbo 🔒	Vanilla	0.605	$0.519_{(-)0.086}$	0.475	$0.565_{(+)0.090}$
	CoT	0.519	$0.506_{(-)0.012}$	0.490	$0.360_{(-)0.130}$
Claude-Sonnet 🔒	Vanilla	0.914	$0.895_{(-)0.019}$	0.800	$0.680_{(-)0.120}$
	CoT	0.907	$0.877_{(-)0.031}$	0.820	$0.730_{(-)0.090}$
LLaMA-3.1-70B-Instruct	Vanilla	0.642	$0.593_{(-)0.049}$	0.750	$0.660_{(-)0.090}$
	CoT	0.870	$0.827_{(-)0.043}$	0.760	$0.720_{(-)0.040}$
LLaMA-3-70B-Instruct	Vanilla	0.673	$0.623_{(-)0.049}$	0.655	$0.495_{(-)0.160}$
	CoT	0.883	$0.809_{(-)0.074}$	0.400	$0.360_{(-)0.040}$
LLaMA-3-8B-Instruct	Vanilla	0.667	$0.617_{(-)0.049}$	0.325	$0.340_{(+)0.015}$
	CoT	0.599	$0.660_{(+)0.062}$	0.375	$0.355_{(-)0.020}$
Mixtral-8x7B-Instruct-v0.1	Vanilla	0.327	$0.401_{(+)0.074}$	0.485	$0.110_{(-)0.375}$
	CoT	0.370	$0.284_{(-)0.086}$	0.395	$0.285_{(-)0.110}$
Mistral-7B-Instruct-v0.3	Vanilla	0.481	$0.537_{(+)0.056}$	0.180	$0.055_{(-)0.125}$
	CoT	0.475	$0.506_{(+)0.031}$	0.200	$0.120_{(-)0.080}$
Phi-3-Medium-128K-Instruct	Vanilla	0.543	$0.568_{(+)0.025}$	0.465	$0.390_{(-)0.075}$
	CoT	0.698	$0.574_{(-)0.123}$	0.325	$0.330_{(+)0.005}$
Phi-3-Small-128K-Instruct	Vanilla	0.580	$0.531_{(-)0.049}$	0.490	$0.520_{(+)0.030}$
	CoT	0.728	$0.568_{(-)0.160}$	0.395	$0.485_{(+)0.090}$
Phi-3-Mini-128K-Instruct	Vanilla	0.420	$0.352_{(-)0.068}$	0.755	$0.665_{(-)0.090}$
	CoT	0.481	$0.370_{(-)0.111}$	0.735	$0.655_{(-)0.080}$

Table 6: All results for **Logic**.

Model	Setting	GSM8K		SVAMP	
		Original	AAVE	Original	AAVE
GPT-o1 🔒	Vanilla	0.953	$0.927_{(-)0.027}$	0.940	$0.920_{(-)0.020}$
GPT-4o 🔒	Vanilla	0.933	$0.947_{(+)0.013}$	0.933	$0.913_{(-)0.020}$
	CoT	0.967	$0.933_{(-)0.033}$	0.933	$0.907_{(-)0.027}$
GPT-4 🔒	Vanilla	0.840	$0.640_{(-)0.200}$	0.840	$0.787_{(-)0.053}$
	CoT	0.947	$0.867_{(-)0.080}$	0.893	$0.760_{(-)0.133}$
GPT-3.5-turbo 🔒	Vanilla	0.587	$0.287_{(-)0.300}$	0.747	$0.600_{(-)0.147}$
	CoT	0.780	$0.480_{(-)0.300}$	0.727	$0.607_{(-)0.120}$
Claude-Sonnet 🔒	Vanilla	0.973	$0.947_{(-)0.027}$	0.967	$0.913_{(-)0.053}$
	CoT	0.973	$0.960_{(-)0.013}$	0.933	$0.920_{(-)0.013}$
LLaMA-3.1-70B-Instruct	Vanilla	0.680	$0.920_{(+)0.240}$	0.853	$0.867_{(+)0.013}$
	CoT	0.867	$0.927_{(+)0.060}$	0.893	$0.813_{(-)0.080}$
LLaMA-3-70B-Instruct	Vanilla	0.933	$0.920_{(-)0.013}$	0.880	$0.853_{(-)0.027}$
	CoT	0.947	$0.907_{(-)0.040}$	0.900	$0.867_{(-)0.033}$
LLaMA-3-8B-Instruct	Vanilla	0.847	$0.800_{(-)0.047}$	0.807	$0.800_{(-)0.007}$
	CoT	0.820	$0.800_{(-)0.020}$	0.833	$0.800_{(-)0.033}$
Mixtral-8x7B-Instruct-v0.1	Vanilla	0.427	$0.193_{(-)0.233}$	0.613	$0.487_{(-)0.127}$
	CoT	0.673	$0.573_{(-)0.100}$	0.700	$0.560_{(-)0.140}$
Mistral-7B-Instruct-v0.3	Vanilla	0.367	$0.147_{(-)0.220}$	0.433	$0.280_{(-)0.153}$
	CoT	0.420	$0.320_{(-)0.100}$	0.487	$0.373_{(-)0.113}$
Phi-3-Medium-128K-Instruct	Vanilla	0.893	$0.833_{(-)0.060}$	0.840	$0.747_{(-)0.093}$
	CoT	0.893	$0.853_{(-)0.040}$	0.827	$0.800_{(-)0.027}$
Phi-3-Small-128K-Instruct	Vanilla	0.840	$0.793_{(-)0.047}$	0.800	$0.727_{(-)0.073}$
	CoT	0.880	$0.873_{(-)0.007}$	0.907	$0.813_{(-)0.093}$
Phi-3-Mini-128K-Instruct	Vanilla	0.520	$0.573_{(+)0.053}$	0.520	$0.527_{(+)0.007}$
	CoT	0.800	$0.807_{(+)0.007}$	0.747	$0.693_{(-)0.053}$

Table 7: All results for **Math**.

times simulates an algorithm to solve math problems after standardization (e.g., '*Let* (x) *be the number of apple pie boxes...*'). However, it does not fully solve the problem in the end and only returns a formula (e.g., '*30x + 255*'), which indicates that the model's reasoning ability is limited when it comes to program simulation for queries expressed in dialects.

A.11 Synthetic Lexical Feature list

Following (Ziems et al., 2022), we implement a feature list with distinct AAVE lexicons: ['got', 'ain't', 'no', "'", 'gonna', 'wanna', 'gotta', 'done', 'been', 'finna', 'gunna', 'gon',], and compute the Spearman's correlation between their frequencies in AAVE ReDial and the performance drop.

Model	Setting	Original	AAVE
GPT-o1 🔒	Vanilla	0.942	$0.925_{(-)0.017}$
GPT-4o 🔒	Vanilla	0.783	$0.312_{(-)0.471}$
	CoT	0.762	$0.662_{(-)0.1}$
GPT-4 🔒	Vanilla	0.217	$0.133_{(-)0.083}$
	CoT	0.283	$0.058_{(-)0.225}$
GPT-3.5-turbo 🔒	Vanilla	0.200	$0.129_{(-)0.071}$
	CoT	0.075	$0.067_{(-)0.008}$
Claude-Sonnet 🔒	Vanilla	0.879	$0.717_{(-)0.162}$
	CoT	0.900	$0.771_{(-)0.129}$
LLaMA-3.1-70B-Instruct	Vanilla	0.392	$0.113_{(-)0.279}$
	CoT	0.579	$0.500_{(-)0.079}$
LLaMA-3-70B-Instruct	Vanilla	0.158	$0.067_{(-)0.092}$
	CoT	0.517	$0.350_{(-)0.167}$
LLaMA-3-8B-Instruct	Vanilla	0.025	$0.067_{(+)0.042}$
	CoT	0.029	$0.025_{(-)0.004}$
Mixtral-8x7B-Instruct-v0.1	Vanilla	0.100	$0.075_{(-)0.025}$
	CoT	0.133	$0.071_{(-)0.062}$
Mistral-7B-Instruct-v0.3	Vanilla	0.096	$0.075_{(-)0.021}$
	CoT	0.083	$0.083_{(-)-0.0}$
Phi-3-Medium-128K-Instruct	Vanilla	0.050	$0.037_{(-)0.013}$
	CoT	0.067	$0.029_{(-)0.037}$
Phi-3-Small-128K-Instruct	Vanilla	0.058	$0.062_{(+)0.004}$
	CoT	0.096	$0.079_{(-)0.017}$
Phi-3-Mini-128K-Instruct	Vanilla	0.021	$0.042_{(+)0.021}$
	CoT	0.017	$0.021_{(+)0.004}$

Table 8: All results for **Integrated**.